Algorithmic Component and System Reliability Analysis of Truss Structures

by

Sepehr Hashemolhosseini

Thesis presented in fulfilment of the requirements for the degree of Master of Science in Civil Engineering at Stellenbosch University

Department of Civil Engineering,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa

Supervisor : Mr. E. Van der Klashorst

December 2013

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Signature: ....

S. Hashemolhosseini

November 26, 2013

Date: ....

Abstract

Most of the parameters involved in the design and analysis of structures are of stochastic nature. This is, therefore, of paramount importance to be able to perform a fully stochastic analysis of structures both in component and system level to take into account the uncertainties involved in structural analysis and design. To the contrary, in practice, the (computerised) analysis of structures is based on a deterministic analysis which fails to address the randomness of design and analysis parameters. This means that an investigation on the algorithmic methodologies for a component and system reliability analysis can help pave the way towards the implementation of fully stochastic analysis of structures in a computer environment. This study is focused on algorithm development for component and system reliability analysis based on the various proposed methodologies. Truss structures were selected for this purpose due to their simplicity as well as their wide use in the industry. Nevertheless, the algorithms developed in this study can be used for other types of structures such as moment-resisting frames with some simple modifications.

For a component level reliability analysis of structures different methods such as First Order Reliability Methods (FORM) and simulation methods are proposed. However, implementation of these methods for the statistically indeterminate structures is complex due to the implicit relation between the response of the structural system and the load effect. As a result, the algorithm developed for the purpose of component reliability analysis should be based on the concepts of Stochastic Finite Element Methods (SFEM) where a proper link between the finite element analysis of the structure and the reliability analysis methodology is ensured. In this study various algorithms are developed based on the FORM method, Monte Carlo simulation, and the Response Surface Method (RSM). Using the FORM method, two methodologies are considered: one is based on the development of a finite element code where required alterations are made to the FEM code and the other is based on the usage of a commercial FEM package. Different simulation methods are also implemented: Direct Monte Carlo Simulation (DMCS), Latin Hypercube Sampling Monte Carlo (LHCSMC), and Updated Latin Hypercube Sampling Monte Carlo (ULHCSMC). Moreover, RSM is used together with simulation methods.

Throughout the thesis, the efficiency of these methods was investigated. A Fully Stochastic Finite Element Method (FSFEM) with alterations to the finite element code seems the fastest approach since the linking between the FEM package and reliability analysis is avoided. Simula-

tion methods can also be effectively used for the reliability evaluation where ULHCSMC seemed to be the most efficient method followed by LHCSMC and DMCS. The response surface method is the least straight forward method for an algorithmic component reliability analysis; however, it is useful for the system reliability evaluation.

For a system level reliability analysis two methods were considered: the β-unzipping method and the branch and bound method. The β-unzipping method is based on a level-wise system reliability evaluation where the structure is modelled at different damaged levels according to its degree of redundancy. In each level, the so-called unzipping intervals are defined for the identification of the critical elements. The branch and bound method is based on the identification of different failure paths of the structure by the expansion of the structural failure tree. The evaluation of the damaged states for both of the methods is the same. Furthermore, both of the methods lead to the development of a parallel-series model for the structural system. The only difference between the two methods is in the search approach used for the failure sequence identification.

It was shown that the β-unzipping method provides a better algorithmic approach for evaluating the system reliability compared to the branch and bound method. Nevertheless, the branch and bound method is a more robust method in the identification of structural failure sequences. One possible way to increase the efficiency of the β-unzipping method is to define bigger unzipping intervals in each level which can be possible through a computerised analysis. For such an analysis four major modules are required: a general intact structure module, a damaged structure module, a reliability analysis module, and a system reliability module.

In this thesis different computer programs were developed for both system and component reliability analysis based on the developed algorithms. The computer programs are presented in the appendices of the thesis.

Opsomming

Meeste van die veranderlikes betrokke by die ontwerp en analise van strukture is stogasties in hul aard. Om die onsekerhede betrokke in ontwerp en analise in ag te neem is dit dus van groot belang om 'n ten volle stogastiese analise te kan uitvoer op beide komponent asook stelsel vlak. In teenstelling hiermee is die gerekenariseerde analise van strukture in praktyk gebaseer op deterministiese analise wat nie suksesvol is om die stogastiese aard van ontwerp veranderlikes in ag te neem nie. Dit beteken dat die ondersoek na die algoritmiese metodiek vir komponent en stelsel betroubaarheid analise kan help om die weg te baan na die implementering van ten volle rekenaarmatige stogastiese analise van strukture. Dié studie se fokus is op die ontwikkeling van algoritmes vir komponent en stelsel betroubaarheid analise soos gegrond op verskeie voorgestelde metodes. Vakwerk strukture is gekies vir dié doeleinde as gevolg van hulle eenvoud asook hulle wydverspreide gebruik in industrie. Die algoritmes wat in dié studie ontwikkel is kan nietemin ook vir ander tipes strukture soos moment-vaste raamwerke gebruik word, gegewe eenvoudige aanpassings.

Vir 'n komponent vlak betroubaarheid analise van strukture word verskeie metodes soos die "First Order Reliability Methods" (FORM) en simulasie metodes voorgestel. Die implementering van die metodes vir staties onbepaalbare strukture is ingewikkeld as gevolg van die implisiete verband tussen die gedrag van die struktuur stelsel en die las effek. As 'n gevolg, moet die algoritme wat ontwikkel word vir die doel van komponent betroubaarheid analise gebaseer word op die konsepte van stogastiese eindige element metodes ("SFEM") waar 'n duidelike verband tussen die eindige element analise van die struktuur en die betroubaarheid analise verseker is. In hierdie studie word verskeie algoritmes ontwikkel wat gebaseer is op die FORM metode, Monte Carlo simulasie, en die sogenaamde "Response Surface Method" (RSM). Vir die gebruik van die FORM metode word twee verdere metodologieë ondersoek: een gebaseer op die ontwikkeling van 'n eindige element kode waar nodige verandering aan die eindige element kode sélf gemaak word en die ander waar 'n kommersiële eindige element pakket gebruik word. Verskillende simulasie metodes word ook geïmplimenteer naamlik Direkte Monte Carlo Simulasie (DMCS), "Latin Hypercube Sampling Monte Carlo" (LHCSMC) en sogenaamde "Updated Latin Hypercube Sampling Monte Carlo" (ULHCSMC). Verder, word RSM tesame met die simulasie metodes gebruik.

In die tesis word die doeltreffendheid van die bostaande metodes deurgaans ondersoek. 'n Ten

volle stogastiese eindige element metode ("FSFEM") met verandering aan die eindige element kode blyk die vinnigste benadering te wees omdat die koppeling tussen die eindige element metode pakket en die betroubaarheid analise verhoed word. Simulasie metodes kan ook effektief aangewend word vir die betroubaarheid evaluasie waar ULHCSMC as die mees doeltreffend voorgekom het, gevolg deur LHCSMC en DMCS. The RSM metode is die mees komplekse metode vir algoritmiese komponent betroubaarheid analise. Die metode is egter nuttig vir sisteem betroubaarheid analise.

Vir sisteem-vlak betroubaarheid analise is twee metodes oorweeg naamlik die "β-unzipping" metode and die "branch-and-bound" metode. Die "β-unzipping" metode is gebaseer op 'n sisteem-vlak betroubaarheid ontleding waar die struktuur op verskillende skade vlakke gemodelleer word soos toepaslik vir die hoeveelheid addisionele las paaie. In elke vlak word die sogenaamde "unzipping" intervalle gedefinieer vir die identifikasie van die kritiese elemente. Die "branch-and-bound" metode is gebaseer op die identifikasie van verskillende faling roetes van die struktuur deur uitbreiding van die falingsboom. The ondersoek van die skade toestande vir beide metodes is dieselfde. Verder kan beide metodes lei tot die ontwikkeling van 'n parallel-serie model van die strukturele stelsel. Die enigste verskil tussen die twee metodes is in die soek-benadering vir die uitkenning van falingsmodus volgorde.

Dit word getoon dat die "β-unzipping" metode 'n beter algoritmiese benadering is vir die ontleding van sisteem betroubaarheid vergeleke met die "branch-and-bound" metode. Die "branch-and-bound" metode word nietemin as 'n meer robuuste metode vir die uitkenning van die falings volgorde beskou. Een moontlike manier om die doeltreffendheid van die "β-unzipping" metode te verhoog is om groter "unzipping" intervalle te gebruik, wat moontlik is vir rekenaarmatige analise. Vir so 'n analise word vier hoof modules benodig naamlik 'n algemene heel-struktuur module, 'n beskadigde-struktuur module, 'n betroubaarheid analise module en 'n sisteem betroubaarheid analise module.

In dié tesis is word verskillende rekenaar programme ontwikkel vir beide sisteem en komponent betroubaarheid analise. Die rekenaar programme word in die aanhangsels van die tesis aangebied.

Acknowledgements

I would like to express my deepest gratitude to the following people:

My supervisor Mr. E. Van der Klashorst whose guidance and support helped immensely in preparing this thesis.

Dr. C. Vilijon and Professor M. Holicky for the course on reliability of structures which was a great motivation for my thesis.

Dr. JAvB Strasheim for the course on finite element analysis of structure.

Strand7 technical support team whose swift responses to my emails helped me in the implementation of Strand7 API in this thesis.

My parents whose support made it possible for me to pursue my studies to a postgraduate level.

Contents

List of Figures

List of Tables

Abbreviations

API	Application Programming Interface
CDF	Cumulative Distribution Function
DCMS	Direct Monte Carlo Simulation
FD-SFEM	Finite Difference based Stochastic Finite Element Methods
FEM	Finite Element Methods
FOSM	First Order Second Moment
FSFEM	Fully Stochastic Finite Element Methods
LHCSMC	Latin Hypercube Sampling Monte Carlo
LSF	Limit State Function
PDF	Probability Density Function
RI	Reliability Index
RSM	Response Surface Method
RSM-DMCS	Response Surface based Direct Monte Carlo Simulation
RSM-FOSM	Response Surface based First Order Second Moment
RSM-LHCMSC	Response Surface based Latin Hypercube Sampling Monte Carlo
SFEM	Stochastic Finite Element Methods
ULHCSM	Updated Latin Hypercube Sampling Monte Carlo

Nomenclature

Variables

A	Element cross sectional area
a_i	influence factor of element i
C	Covariance matrix
d'	vector of degrees of freedom
E	Load effect
E_0	Structural response for $d = 0$
E_M	Modulus of elasticity
F_k	cumulative distribution function of random variable X_k
f_y	yield stress
G	Performance function
J	Jacobian matrix
K	Element global stiffness matrix
k	effective length coefficient
m	Unbiased estimation of mean value
N_f	Number of failures in Monte Carlo simulation
P	Permutation matrix
P_f^T	True probability of failure
R	Resistance

r radius gyration

s Unbiased estimation of standard deviation

T Spearman matrix

U_e Strain Energy

X_k Random variable X at its characteristic value

k' Element local stiffness matrix

P_f Probability of failure

$P_{f_{sys}}$ System probability of failure

Greek Symbols

α_i Sensitivity factor corresponding to random variable X_i

α_X Skewness of the Probability density function of random variable X

β_S^i System reliability at level i

β Reliability index

μ_X^e Equivalent normal mean value of a non-normal probability distribution

Φ_U^{-1} Inverse of cumulative distribution function of random variable X

σ Stress

σ_X^e Equivalent normal standard deviation of a non-normal probability distribution

θ element orientation angle

$\varphi(x)$ Probability density function of random variable X

w coefficient of variation

β_P Reliability index of a parallel system

μ_X Mean value of the probability density function of random variable X

Φ_n Multivariate normal distribution function

Φ_X Cumulative distribution function of random variable X

ρ_{XY} Correlation coefficient between variables X and Y

σ_X Standard deviation of the probability density function of random variable X

G^e Equivalent linear safety margin

Chapter 1

Introduction

1.1 Background

Design of safe structures has always been an issue in the field of structural engineering. When the structural elements are designed they have to satisfy certain criteria under the applied loads. It means the violation of the design criteria will be unacceptable, and the structural component will be considered in a failure state. In other terms, the response of the structure is a function of the applied loads, structural strength, and stiffness.

The problem regarding the design of safe structures is owing to the fact that parameters involved in the safety of structures are subject to uncertainties. The uncertainties include: randomness of actions, material properties, geometrical data, uncertainties of theoretical models due to the simplification of actual condition, etc... [22]. In order to address the uncertainties in the design process of the structures, different methodologies have been proposed. These methods are ranging from the method of permissible stresses to the "most recently" proposed methods such as partial safety factors or Load and Resistance factor methods (LRFD). Although the latter two methods are probabilistic design methods, they only consider the probabilistic characteristics of the aforementioned parameters to define the load and resistance factors which are used together with a deterministic analysis [5]. In fact, after the determination of the load and resistance parameters all of the factors involved in structural design (or assessment) are treated as deterministic variables [42]. Conversely, as mentioned above, most of these factors are random and hence stochastic in essence. This will lead to the need for a complete stochastic analysis and assessment of the structure. Recent failures of structures analysed based on these deterministic methods places great emphasis on the stochastic methods as opposed to the current deterministic methods. The failure of I-5 Skagit River Bridge and the I-35W bridge as well as many others are examples of this fact [20]. Figures 1.1 and 1.2 show these structure collapses.

In short, the current deterministic practices don't seem to be as effective to ensure the safety of the structures. It is, therefore, essential that the course for structural design be shifted towards the stochastic methods that can more efficiently address the inevitable randomness and uncertainties involved in structural design and assessment.

Figure 1.1: Failure of Bridge I-35 Mississippi River bridge

Figure 1.2: Failure of I-5 Skagit River bridge

1.2 Aim and scope

The aim of this study is to investigate the proposed methods of reliability analysis both in component and system level as well as their applicability, efficiency and appropriateness for the algorithmic system and component reliability analysis. The thesis is focused on the fully stochastic methodologies of structural assessment. Different methods of component and system level structural reliability are investigated and proper algorithmic methods and computer programs are developed for the proposed methodologies. Important issues regarding the algorithmic component and system level reliability analysis of truss structures are to be investigated and addressed. It should also be noted that the concepts of Stochastic Finite Element Methods (SFEM) are utilised in the development of algorithms and computer programs. The aims of the thesis can be outlined as stated below:

- Investigation of the methods of component reliability analysis of indeterminate truss structures addressing the issue of implicit limit state functions. Methods such as sensitivity based FORM, Monte Carlo simulation, and response surface method are to be investigated.

- Comparing different methods of component reliability analysis as well as comparing different approaches within a methodology such as comparison between various sensitivity analysis methods or various Monte Carlo simulation methods.
- Investigation of the effect and importance of FEM analysis on different methods of component reliability analysis.
- Investigation of the applicability of a commercial finite element software package for a component level reliability analysis using the proposed methods.
- Investigation of the effect of considering the slenderness coefficient that further decreases the capacity of compression members as a deterministic or non-deterministic variable with respect to the yield stress on the element reliability index.
- Investigation of system reliability analysis methods such as the β-unzipping method and the branch and bound method and comparison between the efficiency of the methods for an algorithmic system reliability analysis.
- Investigation of structural failure mode identification using the proposed methods and comparison between the functionality of the proposed methods in structural failure mode identification.
- Investigation of the applicability of element failure modes (tension or compression) in the process of system reliability investigation and failure mechanism identification.
- Investigations and suggestion for the development of integrated computer environment for the system and component reliability analysis of truss structures.

This study is limited to truss structures where all the connections are assumed to be hinged. This means the members of the truss act as bar elements which can only carry axial loads; therefore, the members are either in tension or compression. Truss structures were selected for the following reasons:

- Truss structures are simpler in terms of analysis, and the only failure modes that are required to be considered are compression and tension failure. This helps simplify the complicated study of a fully stochastic structural analysis as well as make it easier to develop algorithmic methodologies for a system and component level stochastic analysis.
- Due to the simplicity of truss structures, they are widely used in construction of bridges, towers, pavilions, etc [26]. As a result, a reliability-based assessment of these structures is essential to ascertain their safety. Not to mention that a reliability-based evaluation can also be utilised to optimize the maintenance and repair costs of these structure [12].

It should also be noted that once the methodologies and algorithms are established, they can be readily generalised to other types of structures with rigid or semi-rigid connections such as frames.

1.3 Layout of the thesis

Chapter two of the thesis is dedicated to the general concept of structural reliability theory. At first, the concepts of different types of failure and their corresponding modes such as ultimate limit state and serviceability limit state that can be used for the reliability evaluation are discussed. Secondly, the definition and the concept of probability of failure and reliability index are presented. Finally, the chapter discusses different types of probability distribution that can be utilised in reliability evaluation of structures as well as throughout this thesis.

After the concepts of reliability index and failure probability are established in Chapter 2 of the thesis, Chapter 3 focuses on the methods of structural reliability analysis. This chapter will mainly discuss the methods that are used for a component level reliability analysis. Initially, the chapter discusses the moment based methods and their development. First Order Reliability Method (FORM) is discussed comprehensively where two different algorithms of FORM reliability analysis are presented. First algorithm is the general procedure that involves solving an equation while the second algorithm is a method based on a Newton-Raphson recursive procedure that is completely numerical. Also a method of computing the equivalent normal parameters for the non-normal random variables is discussed which is used in the FORM algorithms. Eventually, different simulation methods for system reliability evaluation are presented. Methods such as Direct Monte Carlo Simulation (DMCS), Latin Hyper Cube Sampling (LHCS), systematic sampling and updated system sampling are demonstrated. The chapter also briefly mentions the integration method for the calculation of failure probability.

Chapter four covers the topic of implicit limit state functions which is the case where the limit state function is not explicitly available in term of the basic input random variables. Firstly, the concept of implicit limit state functions is discussed. Next, the methodologies to evaluate the reliability of implicit limit states functions are discussed. Methods such as Monte Carlo simulation, response surface method, and sensitivity analysis methods are discussed. Finally, the stochastic finite element formulation that is required for sensitivity-based methods is presented.

Different methods of a component level reliability evaluation for truss structures are investigated in Chapter 5. This chapter covers different methods of performing component level reliability evaluation, and development of algorithms and computer programs for this purpose. Initially, stochastic finite element based methods are used for both a fully stochastic finite element analysis and a finite difference based stochastic finite element analysis where the use of commercially available finite element packages for reliability evaluation is investigated. Next, the response surface method is utilised to tackle the issue of implicit limit state functions, and as a method of forming an explicit limit state function of structural components. Different reliability evaluation methods are implemented together with the response surface method to evaluate the obtained component limit state functions. Simulation based methods are also investigated. This includes crude Monte Carlo simulation, and other methods to improve simulation such as Latin Hypercube sampling method, as well as an updated Latin Hypercube sampling method. Finally, all the methods and their efficiencies are compared. It should be noted that for all the investigated methodologies computer programs are developed.

Chapter 6 of the thesis is a study on methods of system reliability analysis of structures. In this chapter various assumptions regarding the system reliability evaluation together with material models for the system reliability are discussed. The focus is on two major methods of system reliability analysis. First, the branch and bound method for system reliability analysis is explained. Second, the β-unzipping method is reviewed in detail where the system reliability of the structure can be defined in different levels. In short, chapter six is a review of system reliability methods that are implemented in chapter 7 for the system reliability evaluation.

In the seventh chapter of the thesis the practicality of the system reliability methods for a computerised system reliability analysis is investigated. This is done through investigation on a statically indeterminate truss structure where two methods of system reliability analysis are investigated: the β-unzipping method and the branch and bound approach. Firstly, the β-unzipping method is used for the system reliability evaluation of an indeterminate (redundant) truss structure. Next, a program is developed for a computerised system reliability evaluation using the β-unzipping method. Finally, the branch and bound approach is applied to the structure using the computer program which is created based on the concepts developed for the β-unzipping method.

Chapter 8 is dedicated to the conclusions and recommendations regarding a computerised stochastic analysis of the truss structures. The findings of the thesis with respect to both system and component level reliability analysis are presented. Also recommendations are provided for the future study in this topic.

Appendix I of the thesis includes the guidelines for using the application programming interface of a commercial finite element software for the component level reliability analysis of the structure. Appendix II of the thesis includes all the computer programs developed in Matlab for the stochastic assessment of the truss structure together with the finite element analysis coding.

Chapter 2

Structural Reliability Theory

2.1 Introduction to structural reliability

As discussed in Chapter 1, many of the factors involved in the safety of a structure are not of deterministic nature. Most of these variables are in fact random variables. It means these factors each follow a certain type of probability distribution. Structural reliability theory is a way to assess if an element in the structure is safe or unsafe considering all the uncertainties and randomness of factors associated with that element. In other terms, through structural reliability it is possible to see if an element in the structure is reliable or not. It is usually hard to find a good definition of reliability [22]. ISO 2394 defines reliability as the ability of a structure to comply with given requirements under specified conditions during the intended life for which it was designed [22]. Using the concepts of reliability theory one will be able to get a reliability value corresponding to a probability of failure that provides an estimate of the safety of an element. In this chapter firstly the general definition of failure will be given. In the following sections the concept of failure probability will be discussed. Next, the concept behind reliability index will be illustrated. Finally, the prevailing statistical models are shown. Throughout this chapter, it is assumed that the reader is familiar with the basic concepts of probability theory.

2.2 Definition of failure

Defining the reliability of an element in a structure is completely dependent on the definition of failure. Different kinds of failure can be considered for an element. Each of these failure types can be assessed separately, and they can give different probabilities of failure or reliability indices. As a matter of fact, each failure can have its own specific limit state function or performance function (sometimes also called the safety margin) which will consequently lead to a specific reliability index corresponding to that failure type. In structural reliability three types of performance functions for a structural component can be defined [37]. These performance functions are discussed below.

2.2.1 Performance functions based on ultimate limit state

In this case, the strength or the resistance of the elements are evaluated. In fact, it is the assessment of the load-bearing capacity of a specified element to the loads acting on that element. These performance functions may include the tensile strength of an element in tension, the buckling capacity of an element in compression, the moment-bearing capacity, shear resistance, local buckling in steel structures and so forth. Each of these types of failure corresponds to a different type of performance function. It is of utmost importance to define which performance functions or failure types are being investigated for the determination of structural reliability.

2.2.2 Performance functions based on serviceability limit state

Serviceability limit states may also be considered in determination of structural reliability. This type of limit state function is basically dependant on the usage of the structure. As an example, the deflection of a certain element may not exceed a certain value (usually a coefficient multiplied by the span length) or the number of cracks developed in a concrete element may not exceed a certain amount. There is some controversy regarding the use of the serviceability limit states in that the limit states are not necessarily tantamount to the collapse of the structure. They are mostly about the psychological effect on the users of the structure. In most cases where the serviceability limit state is being studied the issue of implicit performance functions arises. This matter will be discussed thoroughly in Chapter 4.

2.2.3 Performance functions based on fatigue failure

Fatigue often becomes an issue for elements that are under cyclic loading. Mostly, due to the cycles of loading and repetition of the applied loads an accumulation of damage and deterioration can take place. This will lead to the collapse of the element. Fatigue can become a serious issue especially on railway bridges. However, the evaluation of the fatigue limit state is out of the scope of this research.

2.3 Probability of failure and reliability index

2.3.1 General concept of probability of failure

The probability of failure of an element is the likelihood of that component to fail. It is usually possible to find the failure probability of an element (or even the whole structure as a system) by stochastically evaluating the performance function of that element relating to the considered failure type.

The variables involved in the performance function or limit state are random variables. It means they follow a specific probability density function. The random variables are shown with capital letters (X) and their realization or sample values are shown with lower-case letters (x). Mostly, the performance function has the form of Equation 2.1 when the ultimate limit state is considered.

$$G = R - E \tag{2.1}$$

In Equation 2.1 R is considered as the resistance of the element and E is the load effect or the load that is applied on the element. Both R and E are in terms of random variables and consequently G is also in terms of those random variables. It can be written in the form shown below. If the random variables are $X_{1,}X_2, ..., X_n$, then:

$$G(X_1, X_{2,}, ..., X_n) = R(X_1, X_{2,}, ..., X_n) - E(X_1, X_{2,}, ..., X_n) \tag{2.2}$$

When the performance function is defined the failure probability can be defined as the probability that $G \leq 0$. It is shown as $P_f = P(G \leq 0)$. There are different methods to calculate this probability of failure which will be mentioned in the following chapters. If the probability distributions of resistance (R) and load effect (E) are known, then they can be depicted as shown in the Figure 2.1 above.

2.3.2 Calculation of failure probability

In most cases both the resistance and the load effect are random variables and accordingly they each follow a certain type of probability distribution (each one has its own probability density function that is usually abbreviated as pdf). In order to calculate the probability of failure, attention should be given to the definition of failure. Based on the definition of failure, failure can happen when the resistance is smaller than the load effect or action. Since the load effect follows a certain probability distribution, then for the failure to happen a realisation of the load effect (e_i) has to be bigger than the resistance (R). Consequently, failure is the summation of all (e_i) values being greater than resistance (R) ($e_i \geq R$) when $E = e_i$. It can be shown as below [37]:

$$P_f = \sum P(E = e_i \cap R < e_i) \tag{2.3}$$

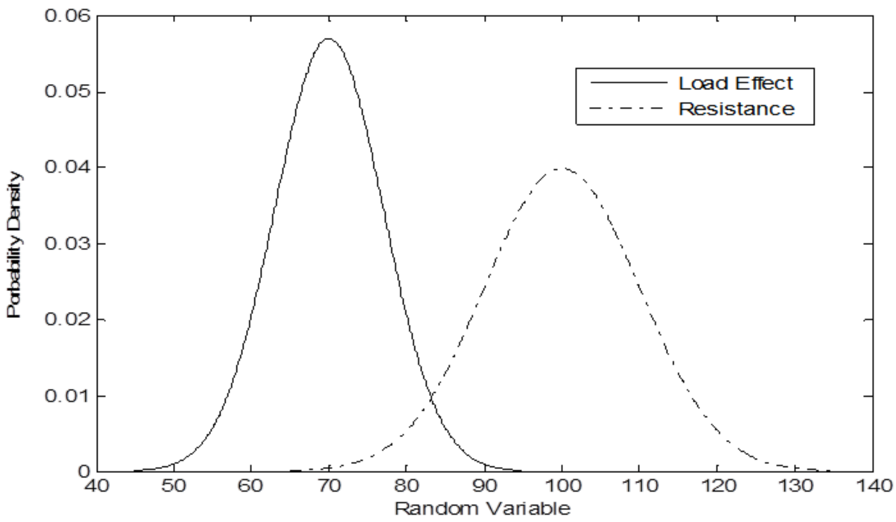

Figure 2.1: Probability density functions of resistance and load effect

Using the concept of conditional probability:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{2.4}$$

So Equation 2.3 becomes:

$$P_f = \sum P(R < E|E = e_i) \times P(E = e_i) \tag{2.5}$$

The integral form of Equation 2.5 can be expressed as below:

$$P_f = \int_{-\infty}^{+\infty} F_R(e_i) \times f_E(e_i) de_i \tag{2.6}$$

Equation 2.6 is used to calculate the probability of failure where F_R is the cumulative distribution function of the resistance and f_E is the probability density function of the load effect. The integral of Equation 2.6 is referred to as the *convolution integral* [32].

Solving the integral shown in Equation 2.6 is important in obtaining the probability of failure; However, closed-form solutions are not always available for this integral [46]. In Chapter 3 proper methods to solve this integral are discussed.

In Figure 2.2 the concept of failure probability is graphically shown. As it is evident from the picture the hatched area is $F_R(e_i)$ and the area between the two vertical lines under the load effect curve is $f_E(e_i)de_i$.

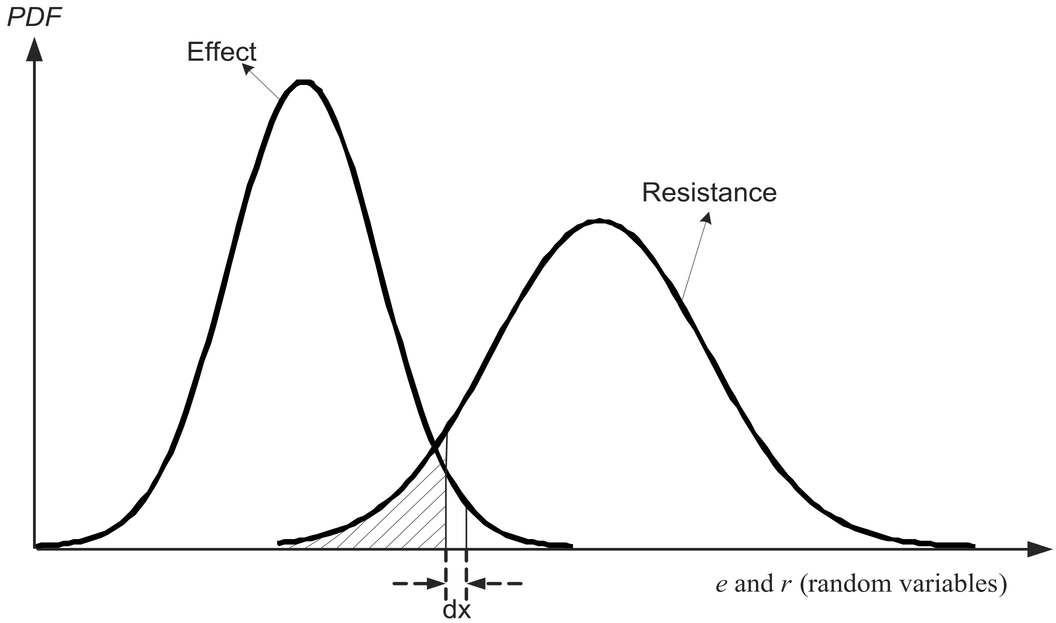

Figure 2.2: Probability density functions of resistance and load effect

2.3.3 Reliability Index

As it was explained in Section 2.3.3, it is possible to calculate a probability of failure according to the probability distributions of the load effect and the resistance. Nevertheless, the concept of failure probability can be shown in a simpler manner. The alternative way to show the safety of a structure is to use the reliability index. In essence reliability index is not different from probability of failure. It is, however, a better way of showing the safety of a structure. In other terms, reliability index provides an easy to understand measure of how safe a structural component or a structure as a whole is. Once the reliability index is calculated, the failure probability can also be calculated. This concept can be shown through a very simple example.

Consider a case where the load effect is deterministic and resistance follows a simple normal distribution. In this case, the load effect has a mean value of 80 and a standard deviation of 0 (since it is deterministic) and resistance has a mean value of 100 and a standard deviation of 10 [22]. Therefore, the calculation of failure probability is as below.

The hatched region in Figure 2.3 is the failure area where the resistance has a smaller value than the load effect. It can be said that if $e = 80$ is considered as a realisation of R, the failure probability will be shown as:

$$P_f = P(R < e) = \Phi_R(e) \tag{2.7}$$

Here Φ_R is the cumulative distribution function of a normal distribution. Usually in statistics the realization of normal distributions can be standardized. If (x) is a realisation of the normal

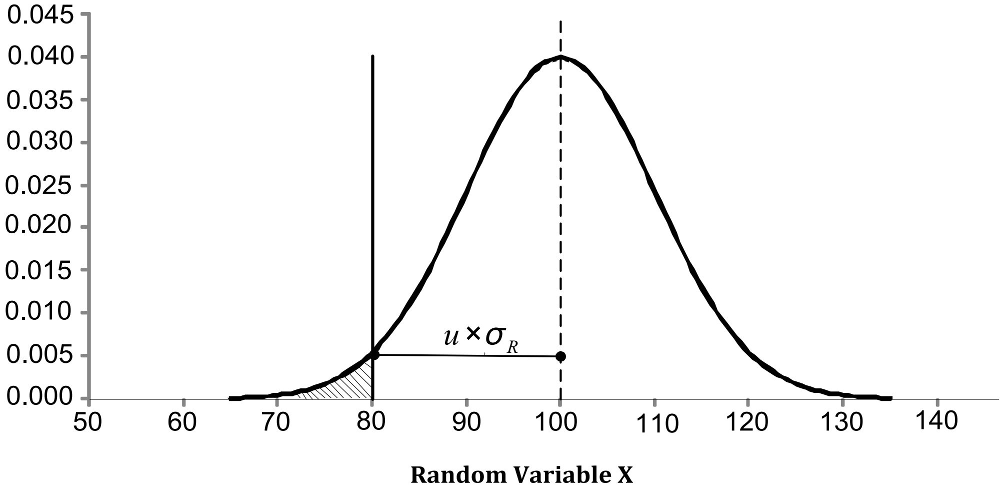

Figure 2.3: Case of deterministic load effect and a normally distributed resistance

random variable (X), the standardised form of (x) can be shown as (u).

$$u = \frac{x - \mu}{\sigma} \tag{2.8}$$

Applying Equation 2.8 to the example:

$$u = \frac{e - \mu_R}{\sigma_R} \tag{2.9}$$

If attention is given to figure 3, it is clear that the distance of the mean value of resistance to the load effect is $\mu_R - e$ which is, from Equation 2.9, equal to $-u \times \sigma_R$. If this distance is expressed considering σ_R as a unit (expressing the distance in terms of standard deviation), then it will have the form of Equation 2.10:

$$- u = \frac{\mu_R - e}{\sigma_R} \tag{2.10}$$

If $-u = \beta$, Equation 2.11 can be shown as below:

$$\beta = \frac{\mu_R - e}{\sigma_R} \tag{2.11}$$

Where β is known as the reliability index.

From this example, one can define reliability index as the distance between the mean of resistance to the mean of load effect [22]. Generally, reliability index can be defined in terms of the failure probability, where Φ_U^{-1} is the inverse of the cumulative function of a standard normal distribution:

$$\beta = -\Phi_U^{-1}(P_f) \tag{2.12}$$

Mostly, both of the resistance R and the load effect E are random variables, so the calculation of reliability index becomes more difficult. In this case, if one can define the probability distribution of G (Equation 2.1), then the probability of failure will be the area where $G < 0$. If a simple case considered when G and R are both normally distributed, stochastic parameters of the probability distribution of G (such as mean value and standard deviation) can be defined as below [22]:

$$\mu_G = \mu_R - \mu_E \tag{2.13}$$

$$\sigma_G^2 = \sigma_R^2 - \sigma_E^2 + \rho_{RE}\sigma_R^2\sigma_E^2 \tag{2.14}$$

If R and E are mutually independent, then $\rho_{RE} = 0$ (coefficient of correlation). Here G will also follow a normal distribution so the probability of failure can be obtained as:

$$P_f = P(G < 0) = \Phi(0) \tag{2.15}$$

This probability can be calculated in the space of the standardised normal random variables. If Equation 2.8 is used to define the standardized normal variables, it is possible to write:

$$u = \frac{0 - \mu_G}{\sigma_G} = -\frac{\mu_G}{\sigma_G} \tag{2.16}$$

With the assumption of G following a normal distribution, $-u$ will be defined as the reliability index which is shown with the symbol (β) . Here, β could be obtained using Equation 2.17

$$\beta = \frac{\mu_G}{\sigma_G} = \frac{\mu_R - \mu_E}{\sqrt{\sigma_R^2 + \sigma_E^2 + 2\rho_{RE}\sigma_R^2\sigma_E^2}} \tag{2.17}$$

Again, if R and E are mutually independent variables, ρ_{RE} will be zero. This concept is shown in Figure 2.4.

In Figure 2.4, the probability of failure is equal to the area of the hatched region (failure area) under the probability distribution function which starts from the point where $G = 0$ (failure boundary). Form the figure, the relationship between the probability of failure and reliability index is evident which means the greater the reliability index is, the less the failure area and the probability of failure is and vice versa. In short, reliability index is just a more conceivable demonstration of failure probability which is not as easy to interpret.

The aforementioned concept of reliability index can also be shown with a 3-D graph in the space of standardised normal random variables as shown in Figure 2.5. The reliability index is defined as the shortest distance from the origin in the space of the standardised normal random variables to the failure boundary ($G = 0$).

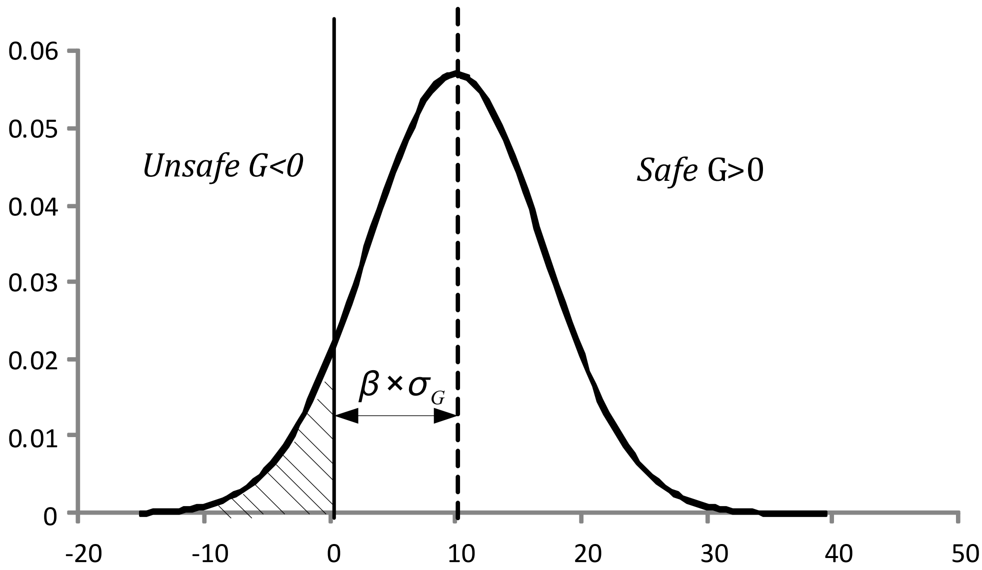

Figure 2.4: Probability density function of G [22]

2.4 Probability distributions of random variables

Most of the parameters related to the issue of structural reliability are not deterministic values, but random variables. If all the likely realisations of a random variable are considered, a population of that random variable can be formed. This population will follow a certain probability model or probability distribution. From the obtained distribution, it is possible to get the likelihood of a random variable having values less than or equal to a certain realisation (for instance, for the yield stress of a certain type of structural steel, one can obtain the probability of yield stress being less than or equal to 250 MPa).

Some probability distributions are significant in the area of structural reliability. In the following sections some of the most important probability models will be discussed and their fundamental parameters will be demonstrated. It should be also noted that only the continuous types of probability distributions are considered since discrete probability distributions are hardly used in this research.

2.4.1 Fundamental parameters of statistical models

If a random variable (X) follows a continuous random probability distribution, the mean value or expected value of X is defined as [33]:

$$\mu_X = \int x\varphi(x)dx \tag{2.18}$$

In Equation 2.18 $\varphi(x)$ is the probability density function of the random variable (X).The second important parameter to be considered is the variance of the random variable. It can be

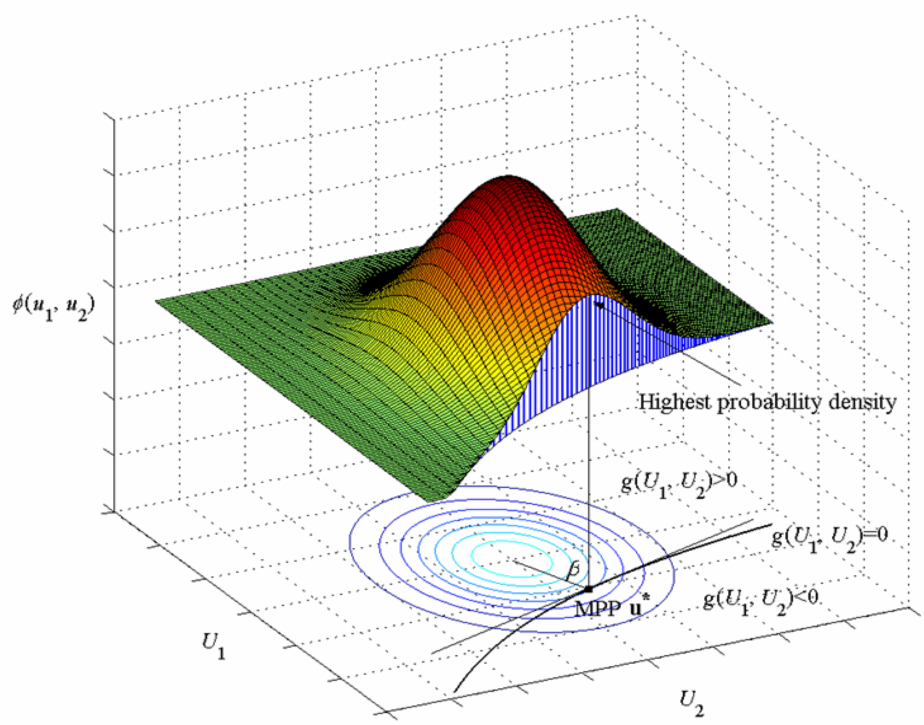

Figure 2.5: Reliability index shown in the space of standardised normal random variables [52]

considered as a measure of variability or dispersion of a random variable:

$$\sigma_X = \int (x-\mu)^2 \varphi(x) dx \tag{2.19}$$

The square root of variance is called the *standard deviation* (σ).

There is another parameter called *skewness* which shows how asymmetric a probability distribution is [22]:

$$\alpha_X = \frac{1}{\sigma^3} \int (x-\mu)^3 \varphi(x) dx \tag{2.20}$$

Another parameter usually used is the *coefficient of variation* which is defined as below:

$$w = \frac{\sigma}{\mu} \tag{2.21}$$

All the parameters defined above, including mean value, variance, and skewness, are referred to as the central moments of the probability distribution. In order to define the probability density function ($\varphi(x)$) first the cumulative distribution function should be defined [22]. Therefore, if the function $\Phi(x)$ is defined as the cumulative distribution function, then it gives the value corresponding to the probability that for a realisation x, random variable X will be less than or equal to x. In a mathematical form:

$$\Phi(x) = P(X \leq x) \tag{2.22}$$

Then one can define the probability density function (pdf) as the derivative of the cumulative distribution function with respect to x:

$$\varphi(x) = \frac{d\Phi(x)}{dx} \tag{2.23}$$

2.4.2 Some important types of random variable distribution

In this section some of the most important distributions that a random variable can have are discussed, and the fundamental parameters of each of these distributions will be presented. Knowing the basic parameters of these random variables is of paramount importance in structural reliability evaluation. It will be demonstrated in the following chapters that in each case of reliability evaluation the fundamental parameters of a certain distribution (such as how load effect is distributed or how resistance is distributed) play a significant role in the resulting value of the reliability index.

2.4.2.1 Uniform random variables

In case of uniform random variables, the probability density function will have a constant value over the range where it is defined. It is in the form of a function such as $f(x) = c$. It means that all the values of random variable have the same likelihood of emergence. If it is defined in the range $[a, b]$, the function will have the value as below:

$$\varphi_X = \frac{1}{a-b} \tag{2.24}$$

If $[X]$ is not in the range $[a,b]$, the value of the function will be zero. The shape of this type of random variable is shown in Figure 2.6 below.

2.4.2.2 Normal random variables

One of the common distributions that can be used for a random variable is a normal distribution. In fact, normal distribution is the most important type of probability distribution in structural reliability [37]. Its probability density function follows the form:

$$\varphi_X(x) = \frac{1}{\sigma_X\sqrt{2\pi}} \exp\left[-0.5\left(\frac{x-\mu_X}{\sigma_X}\right)^2\right] \tag{2.25}$$

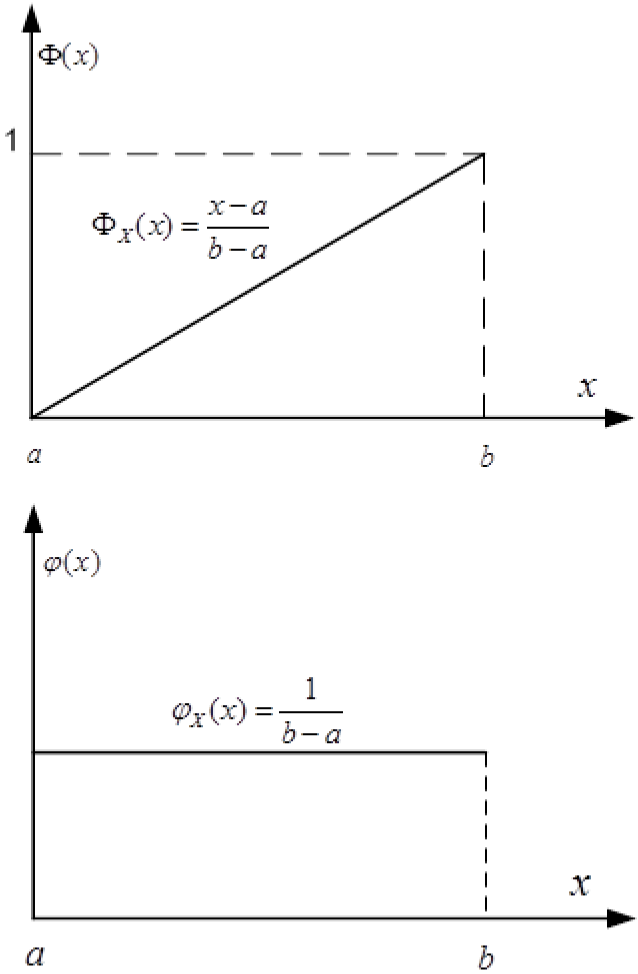

Figure 2.6: CDF and PDf of a uniform distribution shown respectively [22]

There is no closed-form of the cumulative distribution function available for normal random variables. As a result, there are usually some tables available that give the probability for the standardised realisations of a normal distribution. Also programs like Excel or Matlab have some built-in functions that can be used for this purpose. Throughout this thesis these two programs are used for the relevant calculations. A random variable can be standardised using Equation 2.8. In this case the standardised random variable will have a mean value of zero ($\mu_U = 0$) and standard deviation of one ($\sigma_U = 1$).

The normal distribution is often used as a theoretical model in structural reliability. It is used to model some load variables like self-weight, mechanical properties such as strength, and geometrical properties [22]. The normal distribution is defined in the interval $[-\infty, +\infty]$, and its shape is symmetric, as is apparent form Figure 2.7. Consequently, the skewness for a normal distribution will be zero.

2.4.2.3 Lognormal random variables

If a random variable (X) is considered, then it is stated that X possesses a lognormal distribution when $Y = Ln(X)$ has a normal distribution. Lognormal distribution can only be defined for positive values (or absolute values). In order to calculate the probability of a realisation of a lognormal random variable, the fact that Y has a normal distribution is utilised. Thus, it is possible to calculate the probability for a realisation of Y by just using the cumulative distribution function of a standardized normal random variable:

$$P(Y \leq y) = \Phi_Y(\frac{y - \mu_Y}{\sigma_Y}) \tag{2.26}$$

In order to use Equation 2.26 the values of μ_Y and σ_Y are necessary which are the mean value and standard deviation of the lognormal random variable Y respectively. It is known that $y = \ln(x)$, hence $\mu_Y = \mu_{\ln(x)}$ and $\sigma_Y = \sigma_{\ln(x)}$. It is suggested that Equations 2.27 and 2.28 be used to get these parameters [37]:

$$\sigma^2_{\ln(x)} = \ln(w_X^2 + 1) \tag{2.27}$$

$$\mu_{\ln(x)} = \ln(\mu_x) - 0.5\sigma^2_{\ln(x)} \tag{2.28}$$

By obtaining the values of mean and variance using these two equations, it is possible to obtain the probability density function of a lognormal distribution which is shown by $\varphi(x)$ [37]:

$$\varphi(x) = \frac{d}{dX}\Phi\left(\frac{\ln(x) - \mu_{\ln(X)}}{\sigma_{\ln(x)}}\right) = \frac{1}{x\sigma_{\ln(X)}}\phi\left(\frac{\ln(x) - \mu_{\ln(X)}}{\sigma_{\ln(x)}}\right) \tag{2.29}$$

The Lognormal distribution is commonly used in structural reliability. This is especially the case for modelling the resistance properties of different materials like steel, concrete, timber and so forth [14, 22]. Sometimes for some methods of structural reliability analysis, it is needed to calculate the equivalent normal distribution parameters of a lognormal distribution. This matter is discussed comprehensively in Chapter 3.

2.4.2.4 Extreme value distributions

Extreme value distributions, as is evident form the name, are used to model the extreme values of a specific phenomenon over time. They are useful in modelling wind load or seismic load and other types of phenomena where the distribution of maximum or minimum values is important [37]. These distributions often follow an exponential function. There are three types of extreme value distribution: extreme Type I (Gumbel distribution), extreme Type II, and extreme Type III (Weibull distribution).

The first type of extreme value is mostly referred to as Gumbel distribution. It is sometimes used to develop a probabilistic model for wind loads. Mathematically, its CDF and PDF are

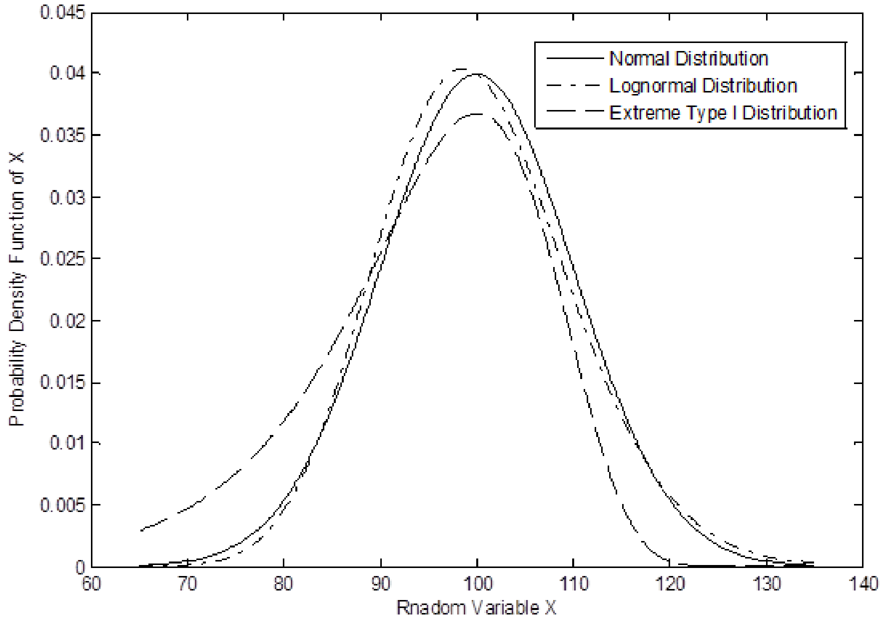

Figure 2.7: Comparison between normal, lognormal and extreme Type I distribution

expressed respectively as below:

$$\Phi_X(X) = \exp(-\exp(-\alpha(x-u)) \tag{2.30}$$

$$u \approx \mu - 0.45\sigma \tag{2.31}$$

The two parameters (α) and (u) are distribution parameters, and they can be calculated in terms of the mean value (μ) and the standard deviation (σ). The approximate equation given underneath can be used to calculate these parameters [37]:

$$\alpha \approx \frac{1.2825}{\sigma} \tag{2.32}$$

$$u \approx \mu - 0.45\sigma \tag{2.33}$$

Extreme type II distribution is addressed as Frechet distribution. It is sometimes used to model the maximum seismic load values that are applied to a structure [37]. Extreme type III is the so-called Weibull distribution. This type of extreme value distribution is beneficial when it is intended to model the minimal values of a certain phenomenon, say material properties such as strength.

Figure 2.7 above shows the difference between normal distribution, lognormal distribution, and extreme Type I distribution for a mean value of 100 and standard deviation of 10.

Chapter 3

Methods of Structural Reliability Analysis

3.1 Introduction to structural reliability

In Chapter 2 the concepts of probability of failure and the reliability index were discussed. In this chapter the methods of reliability analysis will be discussed. There are different methods to evaluate the reliability of a component when its limit state function (safety margin) is known. Each of these methods has its own strengths and weaknesses. Some of the most useful and important methods will be discussed. At the end, a case will be made about what method to choose and why to choose a specific reliability method. An algorithm will be developed to do a computerized component-level reliability analysis of a structure. The focus will be on simulation methods like Monte Carlo Simulation and first order reliability methods like FORM. These methods are discussed in the following sections below.

3.2 First Order Reliability Methods (FORM)

First order reliability methods are one of the common and effective ways of reliability analysis [22]. These methods were developed based on the so-called "second moment" methods that were mainly developed by Cornell [9]. Second moment methods as the name suggests use the information on first and second moments of the random variables to calculate the reliability index. In this section, firstly the first-order second moment method (FOSM) is discussed (Section 3.2.1). Next, the extensions to this method are discussed in Sections 3.2.2 and 3.2.3.

3.2.1 First-Order Second Moment Method (FOSM)

If a linear limit state function is assumed as below [37]:

$$g(X_1, X_2, ..., X_n) = a_0 + a_1 X_1 + ... + a_n X_n \tag{3.1}$$

The reliability index can be calculated as shown in Equation 3.2:

$$\beta = \frac{a_0 + \sum\limits_{i=1}^{n} a_i \mu_i}{\sqrt{\sum\limits_{i=1}^{n} (a_i \sigma_i)^2}} \tag{3.2}$$

It is assumed that all the random variables are uncorrelated. Evidently, the reliability index that is calculated using Equation 3.2 is only in terms of the first two moments of random variables distributions. In fact, it follows the simple rule of adding or subtracting two probability distributions. It is the same concept which is seen in Equation 2.17 of Chapter 2. This expression is only useful and precise when random variables are normally distributed and uncorrelated. In the case where the distributions are not normal, Equation 3.2 only gives a rough approximation of the reliability index [37].

The method can also be used when the limit state function is not linear. In this case the limit state function needs to be linearised. One method to do the linearisation is to use the Taylor Series expansion. If the Taylor Series is only expanded up to the derivative of order one, then it will lead to a linear function. If a function $g(x, y, z)$ is considered, the Taylor Series expansion of $g(x, y, z)$ evaluated at point (a, b, c) will be:

$$g(a, b, c) = (x - a)\frac{\partial g}{\partial x} + (y - b)\frac{\partial g}{\partial y} + (z - c)\frac{\partial g}{\partial z} \tag{3.3}$$

Where $\frac{\partial g}{\partial x}$ is evaluated at point a, $\frac{\partial g}{\partial y}$ is evaluated at point b, and $\frac{\partial g}{\partial z}$ is evaluated at point c.

Using this method, it is possible to linearise a function of n variables in the same manner. One good evaluation point for the case of reliability analysis using FOSM method is to use the mean value of each random variable for the Taylor series expansion. Once the linearisation is done, simply by using Equation 3.2 the reliability index can be calculated. There is also a method available in the literature to do calculations using the FOSM method in case both resistance and load effect follow a lognormal distribution [41]. This method, however, will not be discussed here since it is not of interest for this thesis.

Overall, the FOSM method is a very rough method of reliability calculation and in many practical cases it won't give appropriate answers which are precise enough to be trusted. In many practical cases not all the variables are normally distributed. Also in some cases the basic random variables are correlated, and as a result the method could not be used. Another problem that is associated with this approach is its reliance on the form of the limit state function. when the limit state

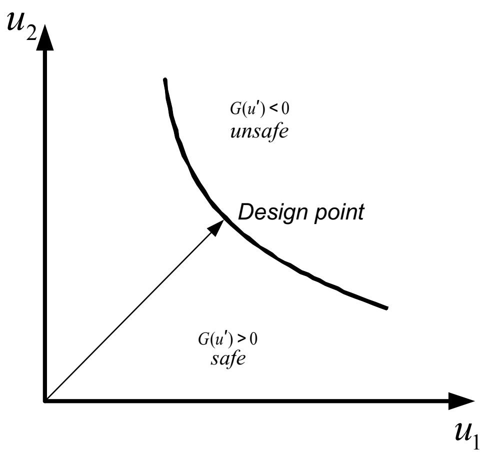

Figure 3.1: Nonlinear limit state Hasofer-Lind reliability index

function is non-linear, it can result in very different reliability indices. The issue is the so-called invariance problem [37]. For instance, different reliability indices are obtained if limit state function is expressed as $R/E < 1$ in lieu of $R - E < 0$ [19]. Moreover, in the linearisation process using the Taylor Series, as formerly mentioned, higher orders of the Taylor Series are ignored for the sake of linearisation. All of the aforementioned problems can demonstrate the need for more sophisticated methods of reliability calculation.

3.2.2 Hasofer-Lind reliability index

The Hasofer-Lind method is a method that was introduced in 1974 [21]. This method was proposed to solve the problem of invariance discussed in the previous section. The Hasofer-Lind method looks at the reliability problem from a geometrical point of view. As a result, the invariance issue will not emerge since it is independent of the way the limit sate function is expressed, as the geometrical shape will always stay the same.

Using this approach, the reliability index will be calculated as the closest distance from the limit state surface to the origin of the coordinate system in the reduced random variable space. Consequently, obtaining the reduced form of the random variables involved in the limit state function is of paramount importance and forms the basis of this method. The transformation to the reduced variable space is simply performed using Equation 2.8.

As is clear from Figure 3.1 above, where the limit state surface is closer to the origin of the random variables, the unsafe area will be greater which means the probability of failure will have a bigger value and thus a smaller value is obtained for β. Looking at the curve in Figure 3.1, one can recognise that the design point is the most probable point on the limit state surface as it is the closest point to the origin of the coordinates system.

The minimum distance from the origin to the limit state surface (β) can be calculated as below.

If (u'^*) is considered as a vector including the coordinates of the design point in the transformed (reduced) coordinate system:

$$\beta = \sqrt{(u'^*)^t(u'^*)} \tag{3.4}$$

In the case of a nonlinear performance function, the problem becomes an optimization problem as it was previously stated in Chapter 2 [19]. The objective of optimization is to minimise the distance from the origin in the transformed coordinated system to the limit state surface to obtain the so-called design point (minimising Equation 3.4). This minimisation is constrained by the fact that $g(U') = 0$ since the point is on the limit state surface. To solve this optimisation problem the concept of Langrange Multipliers can be used [19]. It will lead to the following expression to calculate the value of β:

$$\beta = -\frac{\sum_{i=1}^{n} {u_i}'^* \left(\frac{\partial g}{\partial U_i'}\right)^*}{\sqrt{\sum_{i=1}^{n} \left(\frac{\partial g}{\partial U_i'}\right)^{2*}}} \tag{3.5}$$

In Equation 3.5, the expression $\left(\frac{\partial g}{\partial U_i'}\right)^*$ is the partial derivative evaluated at the design point where the coordinates are $({u_1'}^*, {u'_2}^*, ..., {u'_n}^*)$. In other terms, it can be said that the design point can be obtained from:

$${u'_i}^* = -\alpha_i \beta \tag{3.6}$$

And α_i can be defined as the directional cosines along the U_i coordinate axes. From Equation 3.5, it is clear that α_i can be calculated as below:

$$\alpha_i = \frac{\left(\frac{\partial g}{\partial u_i'}\right)^*}{\sqrt{\sum_{i=1}^{n} \left(\frac{\partial g}{\partial U_i'}\right)^{2*}}} \tag{3.7}$$

Again, here, the terms with superscript asterisks mean that the expression is evaluated at the design point. The value of the design point in the original space using Equation 2.8 is as is shown underneath:

$$x_i^* = \mu_{X_i} - \alpha_i \beta \sigma_{X_i} \tag{3.8}$$

It is clear from the process explained above that the Hasofer-Lind method is an iterative method. The reason for this is that all the calculations are based on the design point coordinates which are not known a-priori. Therefore, the procedure starts with an initial guess, and the values are changed at each of the iterations until the convergence is reached.

One procedure to find the design point is the one proposed by Rackwitz [38]:

1. Determine the appropriate limit state function.

2. Make a preliminary guess for the design point. Usually it is recommended to use the mean values of the random variables as the preliminary guess.

3. Calculate the transformed random variable using Equation 2.8.

4. Calculate the derivative $\frac{\partial g}{\partial U_i'}$ at the design point as well as the directional cosine α_i.

5. Use Equation 3.6 to obtain the design point $u_i'^*$ in terms of reliability index β.

6. Substitute the value obtained in the previous step in the limit state function (LSF), And solve it to find β (note that LSF is in terms of β, and by solving the LSF for β the reliability index is obtained).

7. Using the β value obtained in the previous step, calculate a new design point by utilising Equation 3.6.

8. Repeat steps 3 to 7 until the reliability index value converges.

The Hasofer-Lind reliability index can be connected to failure probability using the following equation:

$$P_f = 1 - \Phi(\beta) \tag{3.9}$$

Here, $\Phi(\beta)$ is the cumulative distribution function of a standard normal distribution.

The problem with the Hasofer-Lind reliability method is that this method will not provide correct information in case the variables are non-normal and independent as well as when the limit state function is not linear [19]. Also step six of this method requires the substitution of the design point value in terms of β into equation $g(u'^*) = 0$ which makes it difficult to perform as far as the computer programming and numerical calculations are concerned. In the next section another method is discussed which was proposed by Rackwitz and Fiessler [39].

3.2.3 FORM method

The FORM method, also known as Rackwitz-Fiessler method, is one of the efficient methods of reliability calculation [22]. In the previous method (Hasofer-Lind method) the information on the type of the distributions were not affecting the calculation and only the mean values and standard deviations were substituted into the equations to get the reliability index. The method proposed by Rackwitz-Fiessler (FORM) takes into account the type of probability density function of the random variables. In this procedure, for the case where the probability density functions are not normal, the so-called equivalent normal mean and standard deviation are calculated. Thus, the type of probability density function of random variables will affect the whole procedure.

In order to calculate the reliability index two different approaches can be used. One approach is to directly solve the limit state equation for β to obtain the value of reliability index. This approach is the same as step 6 of the Hasofer-Lind procedure. The other approach uses a Newton-Raphson recursive process to determine the safety or reliability index instead of directly solving the limit state for β. With respect to the numerical calculation and computer programming, the latter is computationally more convenient since it utilises an iterative trial and error process. This approach is discussed in Section 3.2.3.3.

3.2.3.1 FORM method based on direct solution of limit state function

This type of FORM method includes the steps explained below [19]:

1. In the first step, a proper limit state function needs to be determined. Limit sate functions can clearly be different based on the type of failure that is considered. That is, serviceability or strength, and so forth.

2. It this step an initial guess has to be made regarding the value of reliability index. A reasonable assumption is a key factor when considering the convergence of the algorithm. It is suggested in the literature that a value of 3 can be a sensible assumption [19].

3. The third step is about making an assumption regarding the initial values of the design point. A good initial guess can be the mean values of the random variables involved in the limit state function [37].

4. At this step the calculation of equivalent normal parameters (equivalent mean and standard deviation) must be done if non-normal random variables exist in the performance function (in most of the practical cases non-normal random variables are applicable). The methods to calculate the equivalent normal parameters will be explained in Section 3.2.3.4. It will be seen that the formulae for calculation of the equivalent normal parameters gives the equivalent normal parameters at the guessed design point, so at each of the iterations of the procedure new equivalent parameters are to be calculated.

5. In step five, the partial derivatives need to be calculated at the design point $(\frac{\partial g}{\partial U_i})^*$. Step five is a very important step in FORM process and in case of a computerized analysis of a large structure the issue of implicit limit state functions can affect this step. This issue is comprehensively discussed in Chapter 4.

6. Computation of the directional cosines is done at this step. The directional cosines are also known as the sensitivity factors, and they are a good estimate of how different random variables can contribute to the reliability of the limit state function under study. The equation below is used to calculate the directional cosines:

$$\alpha_{X_i} = \frac{(\frac{\partial g}{\partial X_i})^* \sigma_{X_i}^N}{\sqrt{\sum_{i=1}^{n} (\frac{\partial g}{\partial X_i} \sigma_{X_i}^N)^{2*}}} \tag{3.10}$$

It should be noted that the standard deviations in the directional cosines formula are normal standard deviations. Hence, for the non-normal random variables the equivalent normal standard deviation needs to be obtained.

7. At this step the new design point will be calculated using the sensitivity factors (directional cosines) calculated in the previous step.

$$x_i^* = \mu_{X_i}^N - \alpha_i \beta \sigma_{X_i}^N \tag{3.11}$$

In the first iteration the first set of directional cosines are obtained in step 6. Thus, with the new design point values the process can be repeated from step 4 to 7 until the convergence criteria are met where the difference between two successive estimates of sensitivity factors is considered. It is suggested that a tolerance level of 0.005 is appropriate at this step [19]. After the convergence criterion is met, then step 8 will be performed.

8. Step eight is only performed when the convergence has taken place in step 7. Here, an updated value will be calculated for the reliability index β. It can be calculated using the fact that the design point has to satisfy the equation $G = 0$. It should also be noted that using Equation 3.11, the random variables at the design point are calculated in terms of β.

9. In step nine, the convergence of the reliability index will be checked. If the convergence has taken place there will be no need to perform another iteration. However, if the convergence has not taken place yet, step 3 to 8 has to be repeated until convergence is reached. A change of less than 0.001 in the value of β is recommended [19].

3.2.3.2 FORM method based on Newton-Raphson recursive procedure

The FORM method illustrated previously leads to solving an equation in its 8$^{\text{th}}$ step. As it was previously mentioned, this can complicate the numerical calculation if a computerised method is desired. The issue can get even more difficult when a complicated nonlinear function is confronted. Furthermore, in many cases the limit state function may not be explicitly available in terms of random variables in the matter at hand.

All the aforementioned issues bring about the demand for a method that solves the limit state equation implicitly for β. The method developed by Rackwitz and Fisseler [39] uses a Newton-Raphson algorithm to solve this issue.

The steps of performing the Rackwitz-Fisseler procedure are outlined below:

1. Determine the appropriate limit state function.

2. Make an initial assumption about the design point values (mean values of the involved random variables are often considered as a good initial guess). Calculate the value of the

limit state function by substituting these values into the limit state expression.

3. Compute the equivalent mean and standard deviation for the non-normal basic random variables. The equivalent normal parameters are calculated at the design point.

4. At this step the design point values have to be transformed into standard normal random variables using Equation 2.8.

5. Calculate the partial derivatives $\frac{\partial g}{\partial X_i}$ of the performance function at the design point.

6. Use the chain rule of differentiation to acquire the derivative of the performance function with respect to each variable in the equivalent standard normal space. Hence, if Equation 3.12 is considered,

$$X_i = X_i' \sigma_{X_i}^N + \mu_{X_i}^N \tag{3.12}$$

Then applying the chain rule of differentiation to Equation 3.12, the derivative of limit state function is obtained as below:

$$\frac{\partial g}{\partial X_i'} = \frac{\partial g}{\partial X_i} \times \frac{\partial X_i}{\partial X_i'} \tag{3.13}$$

Therefore:

$$\frac{\partial g}{\partial X'_i} = \frac{\partial g}{\partial X_i} \sigma_{X_i}^N \tag{3.14}$$

Consequently, the directional cosines can be calculated:

$$\alpha_i = \frac{\left(\frac{\partial g}{\partial X_i}\right)^* \sigma_{X_i}^N}{\sqrt{\sum_{i=1}^{n} \left(\frac{\partial g}{\partial X_i}\right)^{2*}}} \tag{3.15}$$

where the superscript asterisks means that Equation 3.15 is evaluated at the design point.

7. Compute the new design point values using the concept of Newton-Raphson optimization (this concept will be briefly discussed in Section 3.2.3.3). It is as given in Equation 3.16:

$$X_{k+1}' = \frac{1}{|\nabla g(X_k'^*)|} [(\nabla g(X_k'^*)^t X_k'^* - g(X_k'^*)] \nabla g(X_k'^*) \tag{3.16}$$

In Equation 3.16, the value of $\nabla g(X_k')$ is the vector of the gradients of the limit state function evaluated at the transformed standardised normal space. This vector can be calculated using Equation 3.14 which will eventually yield a new vector including the new design point.

8. With the obtained values of the new design point the new reliability index can be calculated using the following equation which, in fact, represents the magnitude of a vector that starts

at the origin of the coordinate system and ends at the design point (Figure 3.1):

$$\beta = \sqrt{\sum_{i=1}^{n} \left(x_i'^{*}\right)^2} \tag{3.17}$$

In this step the convergence criterion can be checked. Evidently, in the first iteration the criterion cannot be checked since there is no previous β value available, and this criterion can only be checked form the second iteration on. Usually it is recommended that the difference between the two successive β values should be less than 0.001 [19].

9. Now, the coordinates of the new design point can be calculated in the original variable space using Equation 3.18:

$$x_i^* = \mu_{X_i}^N + \sigma_{X_i}^N x_i'^{*} \tag{3.18}$$

Having computed the new design point values in the original space, it is possible to get the value of the limit state function. Clearly, the design point has to be a point on the limit state surface where $g = 0$. Thus, a tolerance level can be considered. A proper tolerance is that the obtained value for g has to be less than or equal to 0.001.

If both of the convergence criteria in step 8 and 9 are met, then the procedure can be stopped. A Matlab program was developed for the above-mentioned algorithm, and the results were tested for a worked example in [37] (refer to Appendix II).

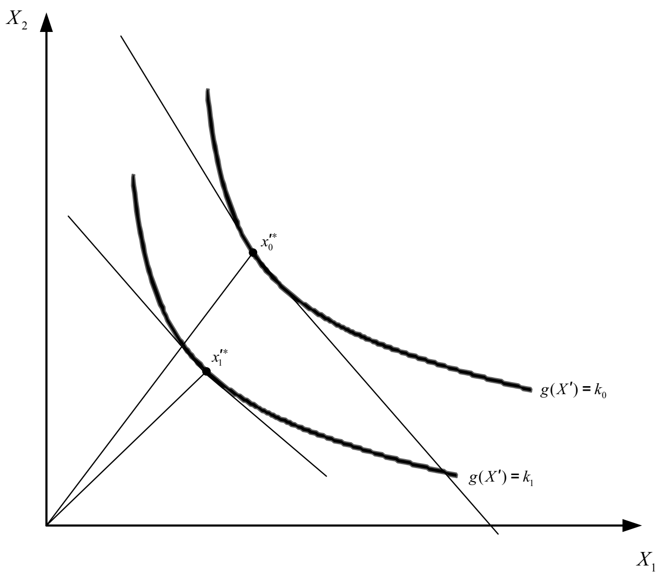

Figure 3.2: FORM method based on Newton-Raphson recursive procedure

3.2.3.3 The recursive Newton-Raphson optimization procedure for the FORM method

It was discussed before that the whole process of finding the reliability index is in fact an optimization process. The procedure in Section 3.2.3.1 for the FORM method will lead to finding the root(s) of the equation $G = 0$ which can be problematic with respect to computation especially when the computation is performed using a computer program. The FORM method described in Section 3.2.3.2 is constructed through the concept which uses a Newton-Raphson recursive procedure for optimization.

Usually, Newton-Raphson is used for finding the roots of equations. However, the concept is also applicable in optimization. When the Newton-Raphson is used to find the root of the first derivative of a function it will consequently yield a stationary point of that function which is clearly a maximal or minimal point of the function.

If x^* is a stationary point of a function $f(x)$, then according to the definition of a stationary point the value of the function should not change in the vicinity of the point x^*. Consider a second order Taylor Series expansion about point x^*:

$$f(x^* + \Delta x) = f(x^*) + f'(x^*)\Delta x + \frac{1}{2}f''(x^*)\Delta x \tag{3.19}$$

where $\Delta x = x - x^*$.

The function in Equation 3.19 will obtain its extremum value if its derivative with respect to Δx is zero. Therefore, it will have the form as shown below:

$$f'(x^*) + f''(x^*)\Delta x = 0 \tag{3.20}$$

So

$$\Delta x = x - x^* = -\frac{f'(x^*)}{f''(x^*)} \rightarrow x = x^* - \frac{f'(x^*)}{f''(x^*)} \tag{3.21}$$

In case of the reliability evaluation of the limit state function the same concept is used. It is based on the linearisation of the performance function recursively. Thus, for the limit state function g, if a first order Taylor series expansion of g is considered:

$$g(x_{k+1}^{'*}) = g(x_k^{'*}) + \nabla g(x_{k+1}^{'*} - x_k^{'*}) \tag{3.22}$$

Here, since the limit state function is in terms of more than one variable, the gradient is used in the Taylor series expansion that leads to Equation 3.22. On the limit sate surface $g(x_{k+1}^{'*}) = 0$; Therefore, Equation 3.22 will be equal to zero and rearrangement of the terms gives Equation 3.16 [19]. This concept is demonstrated in Figure 3.2 above.

3.2.3.4 Normal tail approximation (equivalent normal parameters calculation)

In the third step of FORM reliability analysis, it is required to calculate the equivalent normal parameters for any non-normal variable. In order to get the equivalent normal parameters of a non-normal distribution the normal tail approximation can be used. Conceptually, if a random variable has a PDF $f_X(X)$ and CDF $F_X(X)$ where μ_X is the mean value and σ_X is the standard deviation, then the equivalent normal PDF and CDF must give precisely the same values as the original PDF and CDF for a value of x^*. In mathematical terms, it is stated as below:

$$F_X(x^*) = \Phi\left(\frac{x^* - \mu_X^e}{\sigma_X^e}\right) \tag{3.23}$$

$$f_X(x^*) = \frac{1}{\sigma_X^e}\varphi\left(\frac{x^* - \mu_X^e}{\sigma_X^e}\right) \tag{3.24}$$

In these two expressions, Φ is the CDF and φ is the PDF of a standard normal distribution. μ_X^e and σ_X^e are the equivalent mean and standard deviation of the approximate normal distribution respectively. x^* is the value of the design point in each iteration of the reliability analysis procedure. From Equation 3.23 and 3.24, it is possible to get μ_X^e and σ_X^e as shown:

$$\sigma_X^e = \frac{\varphi(\Phi^{-1}(F_X(x^*)))}{f_X(x^*)} \tag{3.25}$$

$$\mu_X^e = x^* - \sigma_X^e(\Phi^{-1}(F_X(x^*)) \tag{3.26}$$

Equation 3.25 and 3.26 give the equivalent normal parameters of a non-normal distribution. These two formulae are included in the iterative algorithm of FORM reliability analysis. It is evident that at each of the iterations these two parameters need to be calculated in that the value of x^* (design point) changes in each iteration.

3.3 Simulation methods

In Section 3.2 second-order methods for computing the reliability index were discussed. Mostly, these methods can only be used when the limit state function is explicitly available, except for the FORM method that uses Newton-Raphson recursive procedure. Another way of calculating the probability of failure and reliability index is using simulation methods. Simulation methods can be used for both explicit and implicit limit state functions. In simulation methods many random samples for each of the basic random variables in the problem are generated in a way that they can represent their true probability distributions and stochastic characteristics. Using each realisation of the random variables in the problem and analysing the problem deterministically will lead to a realisation of the problem itself. Each of these deterministic analyses is referred to as a *simulation trial*. A large number of simulation trials give the overall stochastic characteristics of the problem, especially when the number of simulation trials tends to infinity [19]. For instance, if the limit state function for a beam with respect to its moment-bearing capacity is formulated where the limit state is in terms of two random variables, By using a simulation method, it is possible to obtain several random samples for each one the random variables according to their probability distributions, and deterministically evaluate the limit state function for each realisation of both of the random variables. This will lead to finding the stochastic characteristics of the limit state itself which can be used to calculate the probability of failure and the reliability index (refer to Section 3.3.6).

Simulation methods are suitable for calculations by a computer [3], and available software products include special subroutines that make it easy to apply simulation methods in a computer environment [22]. As a result, simulation methods provide an appropriate tool to verify the results obtained through other reliability methods (such as FORM) or confirm the validity of the results obtained by a new method.

The Monte Carlo simulation is one of the most common methods of simulation that follows the concepts explained above. It's based on generating random samples according to the probability distributions of the basic random variables which are treated as inputs. These inputs are used for the deterministic analyses of the limit sate function. The obtained results are assessed to get the probability of failure and reliability index. This procedure is explained in detail below. In Sections 3.3.1 to 3.3.5 different sampling methods are discussed. In Section 3.3.6 the calculation of failure probability and reliability index is explained (the approach to evaluate the obtained results for the limit state). Finally, in Section 3.3.7 the method to check the accuracy of the obtained results is shown.

3.3.1 Generation of random values (inverse transform sampling)

The generation of random numbers is possible if a random number generator which can generate uniformly distributed random numbers between 0 and 1 (the probability of the occurrence of all the random numbers between 0 and 1 is equal) is available. Computer programs like Matlab have the ability to generate uniformly distributed random numbers between 0 and 1. Also, in the available literature, tables are provided that give random numbers in this interval. When a random number u_i is generated, this number can be equated to the cumulative distribution function of the random variable as shown in Equation 3.27.

$$u_i = F_X(x_i) \rightarrow x_i = F_X^{-1}(u_i) \tag{3.27}$$

It can be demonstrated as a two-step procedure:

1. Generate a uniformly distributed random number (u_i) in the range [0, 1].
2. Use Equation 3.27 to calculate the sample value (x_i).

This algorithm is very general, and given the accessible functions in a programming language like Matlab can be easily implemented for a Monte Carlo simulation. Throughout this thesis, when the inverse transform sampling is used for generating random samples for the basic random variables, the Monte Carlo method is referred to as the direct Monte Carlo simulation (DMCS).

3.3.2 Latin Hypercube Sampling (LHCS)

In Latin hypercube sampling the probability density function of each random variable is divided into different partitions, and then from each of these "division" a representative value is selected. The representative values are combined in a way that each representative value is used once and only once in the simulation. The following steps can be applied for evaluation of the limit state function (G) [37]:

$$G = G(X_1, X_2, ..., X_j) \tag{3.28}$$

1. Divide the probability density function of each of the random variables into (n) intervals in the considered range (usually $[-\infty, +\infty]$). Partitioning should be performed in a way that the probability of X_i being in the interval is $^1/_n$.

2. Choose a representative value from each interval. This representative value has to be randomly selected; nevertheless, if the number of intervals is large the middle value of the interval can be selected as well.

3. After steps 1 and 2 are performed there will be (n) representative values for each of the random variables in the limit state function. The objective is to create (n) combinations so that each representative value is only used once (it should be noted that there would be n^j combinations available if each value was not supposed to be used only once). For instance, if there are two random variables X_1 and X_2 together with n intervals, then the vectors of representative values will be as below:

   $$V_1 = \{x_{11}, x_{12}, ..., x_{1n}\}$$

   $$V_2 = \{x_{21}, x_{22}, ..., x_{2n}\}$$

   Where x_{jn} in vector V_j is the n^{th} representative value of the j^{th} random variable.

4. In order to create n combinations from the representative values, j random permutations of $p = 1, 2, ..., n$ can be created, and then components can be called form each vector of representative values to create the combinations. For instance, consider a limit state function $G = X_1 - X_2$, if it is intended to have 5 random samples for each of the random variables, 5 intervals are considered in a way that they divide the probability density function of each of the random variables into 5 partitions so that the probability of random variable X_i occurring in each interval is $1/5$. Then from each of the intervals a representative

value is randomly selected. The vectors of representative values can be shown as below.

$$V_1 = [x_{11}, x_{12}, x_{13}, x_{14}, x_{15}]$$

$$V_1 = [x_{21}, x_{22}, x_{23}, x_{24}, x_{25}]$$

Next, two random permutation of $p = 1, 2, ..., 5$ are created which could be $P_1 = [3, 5, 1, 2, 4]$ and $P_2 = [1, 2, 3, 4, 5]$.Therefore, the first combination that is used to evaluate G is composed of the third component of the representative vector V_1 and the first component of the representative vector V_2. This way all of the n combinations can be created. In most of the computer software packages there are ways of randomly permuting numbers (such as "randperm" in Matlab).

5. Eventually, the limit state function G can be evaluated for each of the combinations of random variables. As a result, an approximation of failure probability will be obtained (refer to Section 3.3.7).

The advantage of Latin Hypercube sampling is that it creates a more representative sampling of the distribution since it ensures a good spread of the samples of each random variable even though the number of generated samples is low [30]. In fact, Latin Hypercube sampling will prevent clustering of non-representative sampled values, and it ensures sampling form the tails of the probabilistic distribution of random variables (for instance, if 1000 samples are to be obtained, one sample is taken from the interval $[-\infty, 0.001]$ and one sample is taken from the interval $[0.999, +\infty]$. Therefore, upper tail and lower tail are included in the sampling (whereas in direct Monte Carlo sampling none, one or many samples maybe drawn). In general, Latin Hypercube sampling gives more accurate results, and the estimates of failure probability will have a lower coefficient of variation [30]. This matter is demonstrated in Chapter 5 where the application of this sampling method for the component-level reliability evaluation of truss structures is investigated.

3.3.3 Systematic sampling

If the limit state function is in terms of (k) random variables:

$$G = G(X_1, X_2, ..., X_k)$$

In order to get (n) sample values for each one of the variables a matrix P can be formed where each column of the matrix includes an independent random permutation of the values $i = 1, 2, 3..., n$. Components of matrix P can be denoted as P_{jk} which corresponds to x_{jk}, and x_{jk} is j^{th} simulated value of the k^{th} random variable in the limit state function. Accordingly, x_{jk} can be obtained as below [32]

$$F_k(x_{jk}) = \frac{2p_{jk} - 1}{2N} \tag{3.29}$$

Using the inverse function of F we get:

$$x_{jk} = F_k^{-1}\left(\frac{2p_{jk}-1}{2N}\right) \tag{3.30}$$

In Equations 3.29 and 3.30 F_k is the cumulative distribution function of the k^{th} random variable.

3.3.4 Updated system sampling

Updated system sampling can be seen as an improvement or modification to system sampling [16]. In system sampling when the matrix of random samples is formed there is a correlation between different columns of the sample matrix. The very existence of this correlation can undermine the efficiency of sampling. In updated system sampling an effort is made in order to reduce the correlation among the columns of the random sample matrix.

Assume that there are (k) random variables in the limit state function and the objective is to produce (n) random samples of each random variable. Using systematic sampling matrix R with n rows and (k) columns is formed which contains the random samples. And matrix P is the permutation matrix that contains (n) permutations of $i = 1, 2, 3, .., n$ in each of its k columns. The correlation between the columns of matrix P can be obtained using the so-called Spearman parameter [16]:

$$T_{ij} = \frac{6\sum_n (R_{ni} - R_{nj})^2}{N(N-1)(N+1)} \tag{3.31}$$

T_{ij} is known as the Spearman parameter between the two columns i and j, and its value is in the range [-1,1] since it's stating the correlation. By finding all the T_{ij} values, matrix T can be formed. Matrix T is symmetric and if all the columns of matrix R are independent matrix T is equal to unit matrix. In order to modify the matrix of random variables, first the Cholesky partition of matrix T should be obtained. Cholesky partitioning can be shown as below:

$$T = QQ^T \tag{3.32}$$

Here Q is the lower triangular matrix. If S is shown as the inverse of Q:

$$S = Q^{-1} \tag{3.33}$$

Using S, it is possible to perform a modification to matrix R utilising Equation 3.33:

$$R_s = R \times S^T \tag{3.34}$$

If Spearman matrix T_s is formed for R_s, it can be observed that T_s is closer to unit matrix compared to T. In other words, the correlation between the columns of the random sample matrix is decreased. As the number of modifications increases, the columns get closer to becoming thoroughly independent. This method is used in Chapter 5 together with Latin Hypercube sampling method. It is hence provided with more details in that chapter.

3.3.5 Hybrid sampling methods

Any combination of the sampling methods mentioned in the above sections can be used to ameliorate the efficiency of the Monte Carlo simulation [28]. For instance, it is feasible to use a combination of Latin hypercube sampling and updated system sampling and so forth.

3.3.6 Calculation of probability of failure using simulation methods

Monte Carlo simulation sample values (random realisations) can be drawn for each of the random variables involved in the limit state function through any of the sampling methods specified earlier. Each of these sample values (realisations) can be substituted in the limit state function, and the limit state function can be evaluated deterministically using these sample values. Failure is when the value of the limit state function is smaller than zero. Therefore, a binary vector can be formed for each evaluation of the performance function. If the computed value of the limit state function is smaller than 0 (failure) the corresponding value in the binary vector is 1 and 0 otherwise. This is shown in Equation 3.35 where (n) is the number of simulations. The summation of components of the binary vector will yield the total number of failures for the limit state function. Accordingly, If N_f is the number of failures (total number of times when $g < 0$ due to substitution of sample values into the limit state function), and n is the total number of simulations, an approximation of the failure probability can be calculated as is shown in Equation 3.36.

$$I(i) = \begin{cases} 0 & G \geq 0 \\ 1 & G < 0 \end{cases} i = 1, 2, 3..., n \tag{3.35}$$

$$P_f = \frac{N_f}{n} \tag{3.36}$$

Another method of calculating the failure probability from Monte Carlo simulation is to calculate the mean value and standard deviation of the limit state function using the acquired evaluation values of the limit state function in each simulation. Then, by using Equation 2.17 the reliability index can be calculated. Having obtained the reliability index, it is possible to calculate the probability of failure:

$$P_f = \Phi(-\beta) \tag{3.37}$$

Equations 3.38 and 3.39 can be employed to obtain an unbiased estimation of mean and variance [22]

$$m = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{3.38}$$

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - m)^2 \tag{3.39}$$

The second method assumes that the limit state function follows a normal distribution. Obviously, this assumption is not true in many cases where the basic random variables are not normally distributed. However, in cases of high reliability index values (low probability of failure) where the first method leads to zero probability of failure, this method can be used to get an estimation of the failure probability and the reliability index.

3.3.7 Accuracy and efficiency of probability estimation with MCS

It should be mentioned that despite the simplicity of direct Monte Carlo simulation, it can be subject to considerable error if the number of simulations is not large enough. There are some methods to determine the accuracy of a crude Monte Carlo simulation or to determine the required number of simulations to reach certain accuracy.

One method to determine the accuracy of a Monte Carlo simulation is to approximate a binomial probability distribution with a normal distribution, and make an estimation about 95% confidence interval of the calculated failure probability. It will lead to Equation 3.40 [19]:

$$e\% = \sqrt{\frac{1-P_f^T}{N \times P_f^T}} \times 200\% \tag{3.40}$$

In Equation 3.40, p_f^T is the true probability of failure and N is the number of simulations. Off course, probability of failure is not known a-priori, but if the objective is to confirm the results of other reliability assessment methods such as FORM, this formula can be useful in determining the number of necessary simulations, since already an estimation of the failure probability is available. Usually in engineering problems, it is recommended to have k million numbers of simulation with k being the number of basic random variables [19].

3.4 Integration Method

Another method of finding the probability of failure is to numerically compute the value of the integral of Equation 2.6.

$$P_f = \int_{-\infty}^{+\infty} F_R(e_i) \times f_E(e_i) de_i \tag{3.41}$$

Different values can be considered for e_i and, consequently, the values of cumulative distribution function of resistance (F_R) and probability density function of load effect (f_e) are computed at e_i where de_i can be considered as an increment of numerical integration. If $f(e_i) = F_R(e_i) \times f_E(e_i)$, and the increment is shown as h, then the probability of failure can be calculated using the following numerical integration formula:

$$P_f = h \times [f(e_1) \times f(e_2) \times .... \times f(e_n)] \tag{3.42}$$

Knowing failure probability P_f, the reliability index can be easily calculated using Equation 2.12. In short, this method is a simple numerical integration of Equation 2.6, and in complicated cases it might be difficult to use.

Chapter 4

Implicit Limit States and Stochastic Finite Element Methods (SFEM)

4.1 Introduction

In Chapter 4 the focus is on the cases of reliability analysis where the limit state functions are not explicitly available in terms of the basic random variables, and the so-called implicit limit state functions are applicable. Different methods are discussed which make it possible to perform reliability analysis where a closed-form expression for the limit state function is not available. In Section 4.2 the concept of implicit limit states is discussed in detail. Section 4.3 describes different methods of reliability analysis where the implicit limit state functions are applicable. Firstly, the response surface method is described which deals with finding an approximate closed-form expression for the desired response of the system. Secondly, Monte Carlo simulation is discussed as another means of reliability calculation for the implicit limit state functions. Finally, the sensitivity based methods are explained which investigate the sensitivity of the system response in terms of input random variables. The sensitivity-based methods form the basis for the stochastic finite element analysis [19]. Some of these sensitivity methods may necessitate alterations in the finite element formulations and some other can be used together with a commercial finite element software package where alterations to the finite element code are not possible. As a result, the sections discussing the stochastic finite element methods will follow the sections describing sensitivity methods. In Section 4.3.4 stochastic finite element methods are discussed in detail.

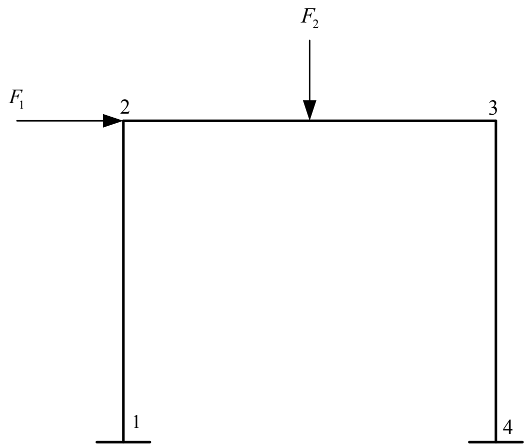

Figure 4.1: Deflection of a portal frame

4.2 Implicit limit state function

An Implicit limit state function is encountered when the limit state function is not explicitly available in terms of the basic random variables in the reliability analysis problem. As a simple example, consider the case when the deflection of a frame or a truss at a certain node is considered. For instance, if a portal frame like the one shown in Figure 4.1 is considered, it is clear that if the maximum allowed deflection of node 3 of the portal frame is 0.9, then the limit state function for deflection can be mathematically expressed as below (Serviceability limit state):

$$g(X) = 0.9 - u_3 \tag{4.1}$$

Looking at Equation 4.1, it is evident that the limit state function is not explicitly available with respect to the input random variables like F_1, F_2, and the resistances of the columns and beams.

Equation 4.1 is an example of an implicit serviceability limit state. The same case applies when the strength limit state function is considered for statistically indeterminate structures where the response of the structure in a certain member or cross section of a structure may not be explicitly available in terms of the input random variables [19].

4.3 Methods of dealing with implicit limit states

4.3.1 Response surface method

The basic concept behind the response surface method is to find a closed-form approximation for the performance function. The approximation can be obtained by performing some deterministic finite element analyses followed by a regression analysis of the results [49]. In this manner, it is possible to get an approximation of the limit state function $g(X)$ by a polynomial expression such as $\hat{g}(X)$ [44]. The obtained approximation of the limit state function will then be analysed by one of the common reliability analysis methods (like FORM) described in the previous chapter. The whole procedure of response surface approach can be summarised as the following steps [19]:

1. If the performance function is in terms of (n) random variables, a set of values has to be selected for each one of random variables. One way to select these sets for the basic random variables is to use factorial experimental design [44]. According to factorial method all the possible combinations should be considered and the performance function has to be analysed for all of these values. For instance, two high and low values like $\mu \pm k\sigma$ can be used for each random variable.

2. The performance function $g(X)$ has to be analysed deterministically for all the sets of values for the random variable in the performance function (deterministic finite element analysis).

3. First-order or second-order regression should then be utilised to obtain a rough closed-form expression for the performance function with respect to the random variables.

4. Any of the methods illustrated in Chapter 3 (such as FORM or Monte Carlo) can be used to do a reliability analysis of the obtained closed-form expression for the limit state function.

4.3.2 Monte Carlo simulation

Monte Carlo simulation is another way of dealing with explicit limit state functions. When using Monte Carlo simulation, a large number of finite element analyses have to be done to calculate the failure probability corresponding to the limit state function. For instance, in the case of the portal frame of Figure 4.1 according to probability distribution of applied loads F_1 and F_2 random numbers can be generated and for each random value a deterministic finite element analysis should be performed to calculate different values for u. Thus, at each deterministic analysis the limit state function is assessed and by using Equation 3.36 the probability of failure can be obtained.

4.3.3 Sensitivity-based methods

Another alternative for dealing with implicit limit state functions is to use any of the sensitivity-based approaches. Using these methods, it is possible to obtain the sensitivity of the structural response with respect to the input random variables. Sensitivity methods are used together with the FORM method. In the FORM analysis only the values and derivatives of the performance function are required at each iteration. The value of the limit state function is computed using a deterministic finite element analysis of the structure, and the derivatives of the performance function with respect to the input random variables are obtained using sensitivity analysis. There are three methods of executing a sensitivity analysis.

4.3.3.1 Finite difference method

Finite difference method is a numerical approach for a rough calculation of the derivatives of a function with respect to its variables. In this method in order to find the derivative in terms of a certain variable, a small change is given to that certain variable while the other variables are at their original values, then the change in the value of the function is calculated. If a function such as $Y = f(X_1, X_2, ..., X_n)$ is considered, where $X_1, X_2, ..., X_n$ are input variables of the function and Y is the output of the function, in order to acquire the derivatives of the function in terms of each of its input variables at a point like $(X_1^0, X_2^0, ..., X_n^0)$ the following procedure can be followed [19]

1. Initially, the value of the function needs to be calculated at the starting point $(X_1^0, X_2^0, ..., X_n^0)$.
2. The value of the variable X_1 should be changed from X_1^0 to $X_1^0 + \Delta X$. Here, ΔX is a very small value and is usually referred to as perturbation in the value of X_1. While the value of X_1 is changed (perturbed), all the other variables are kept at their initial values, and the new value of the function is calculated.
3. In the third step the change in the output value of the function (value of Y) should be investigated. It can be shown as $\Delta Y = Y_1 - Y_0$. Now it is possible to get an approximation for the derivative of the function in terms of variable X_1 which is $\frac{\Delta Y}{\Delta X_1}$.
4. Step 1 to 3 has to be repeated for each variable in the function; however, each time the other variables should be kept at their initial values.

The advantage of finite difference method is that it can be used when the access to the finite element code is not available. Therefore, it can be used with commercial finite element analysis packages.

When using this method inside the FORM analysis the objective function is the limit state function under study. Therefore, for the limit state function the derivatives with respect to each of those variables has to be calculated using this three-step finite difference method. The finite difference method is a very efficient method when a stochastic finite element analysis is done by means of a finite element software package. In the following section it will be explained how a methodology is developed to use the commercial finite element package Strand7 version 2.4.2 to do the stochastic finite element reliability analysis of a truss structure.

4.3.3.2 Classical perturbation

In classical perturbation method the variation of the response is evaluated by checking the variation at every step in terms of the variation of basic variables. This method simply uses the chain rule of differentiation to calculate the derivatives of the performance function with respect to the basic random variables.

4.3.3.3 Iterative perturbation method

This method is used when the non-linear behaviour of the structure is considered. It is based on the fact that during non-linear analysis the response of the structure is obtained through an iterative procedure. Therefore, at each one of the iterations the sensitivity to the basic random variables has to be evaluated. This method will not be used since the thesis will focus on linear static finite element analysis of the structure.

4.3.4 Stochastic Finite Element Methods (SFEM)

In Section 4.3.3 different sensitivity-based methods were discussed. These methods are used to find the derivatives of the performance function in terms of basic input random variables. However, some of the input random variables are used in the finite element formulation of the structure. Thus, in order to apply the sensitivity-based methods to calculate the derivatives of the limit state function, it is either needed to make changes to the finite element formulation or use the finite difference sensitivity method to circumvent the need for changing the finite element formulation. Nevertheless, in the available SFEM literature both of the approaches are considered as a stochastic finite element method since they both use a sensitivity-based method within a FORM algorithm to calculate the reliability index [19, 44]. In the literature, SFEM is defined as a combination of sensitivity analysis and finite element method which leads to a probabilistic analysis [19]. Therefore, In Sections 4.3.4.1 to 4.3.4.3 the general stochastic matrix formulations for truss structures are shown (the programming code for this method is provided in Apendix II of the thesis). In Sections 4.3.4.4 the SFEM method using finite difference method is explained. It should be noted that either of these methods can be used for the reliability calculation. However, since the objective of this study is to provide a comprehensive investigation on reliability analysis of truss structures both of the methods are discussed here. It also provides a good comparison between the two methods. It is important to mention that all of the finite element matrix formulations are for a linear static analysis.

4.3.4.1 Matrix formulation for a linear static Stochastic Finite Element Method

In this section, a matrix approach is discussed that can be used for a stochastic finite element analysis applicable to FORM reliability analysis. In this case, classical perturbation is used as

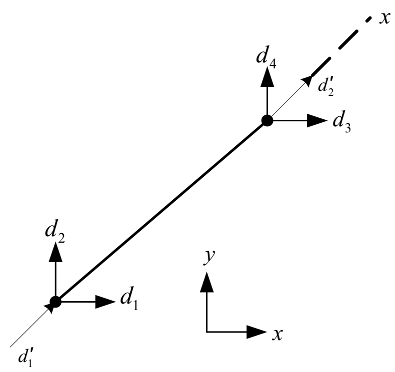

Figure 4.2: Bar element of truss structure

a sensitivity method to calculate the derivatives of the limit state function. Since the focus of the thesis is on truss structures, firstly the deterministic finite element analysis of the truss structures is briefly discussed. Next, the stochastic finite element formulation applicable to truss structures is explained. The presented matrix method will be used to calculate the reliability of the components of a truss structure in Chapter 5.

4.3.4.2 Deterministic Finite Element Analysis of Truss Structures

Here the deterministic finite element analysis of a planar truss structure is briefly discussed [8], since the stochastic finite element analysis is simply based on the deterministic FEM formulation. However, the details are not mentioned in that there is abundant literature available on deterministic finite element methods.

A bar element is considered as shown in Figure 4.2. The local axis of the bar is x', and $x - y$ represents the global axis. Therefore, the vector of degrees of freedom in local axis is:

$$d' = \{d'_1, d'_2\} \tag{4.2}$$

The transformation of the displacement to the global degrees of freedom is demonstrated as:

$$d' = T \times d \tag{4.3}$$

Where T is shown as:

$$T = \begin{bmatrix} c & s & 0 & 0 \\ 0 & 0 & c & s \end{bmatrix} \tag{4.4}$$

In matrix T, $c = \cos\beta$ and $s = \sin\beta$, and β is the angle form x axis.

In order to get the stiffness matrix of a truss element the stiffness matrix of a bar element k'

can be used.

$$k' = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{4.5}$$

The strain energy of the bar can be obtained:

$$U_e = \frac{1}{2} d'^T k' d' \tag{4.6}$$

Substituting $d' = T \times d$ in the strain energy equation:

$$U_e = \frac{1}{2} d^T (T^T k' T) d \tag{4.7}$$

Therefore, the global stiffness can expressed as below:

$$K = T^T k' T \tag{4.8}$$

And the global stiffness matrix for the bar element is obtained as in Equation 4.9 below:

$$K = \frac{EA}{L_e}\begin{bmatrix} c^2 & cs & -c^2 & -cs \\ cs & s^2 & -cs & -s^2 \\ -c^2 & -cs & c^2 & cs \\ -cs & -s^2 & cs & s^2 \end{bmatrix} \tag{4.9}$$

In the reliability analysis, if strength failure is considered, the stresses at elements are the desired response of the system. As a result, the formulation to determine the stresses in truss components needs to be demonstrated as well. Stress in local coordinates is shown as:

$$\sigma = \frac{E}{L_e}\begin{bmatrix} -1 & 1 \end{bmatrix}\begin{bmatrix} d'_1 \\ d'_2 \end{bmatrix} \tag{4.10}$$

Substituting $d' = Td$, yields:

$$\sigma = \frac{E}{L_e}\begin{bmatrix} -c & -s & c & s \end{bmatrix} d \tag{4.11}$$

The global stiffness matrix and the way to calculate element stresses were shown above. For a truss structure the system stiffness matrix should be assembled and Equation 4.12 has to be solved for the displacements. With the displacements known, it is possible to use Equation 4.11 to get the response of the system at component level.

$$F = K \times d \tag{4.12}$$

4.3.4.3 Stochastic finite element formulation and reliability analysis

In order to apply the sensitivity-based approaches to calculate the derivatives of an implicit limit state function within a FORM algorithm, it may be required to change the finite element formulation with respect to the basic random variables (except for the approach that is discussed in Section 4.3.4.4). Accordingly, it has to be illustrated along with the reliability assessment method. In this section, the objective is to use the FORM reliability analysis that uses Newton-Raphson recursive procedure to find the most probable point (design point) and the corresponding reliability index. The steps for application of this method are presented in this section. It should also be mentioned that the matrix formulations for the stochastic finite element analysis are well discussed and developed in the available stochastic finite element literature, and all of the basic matrix formulation in this section are taken from the available literature [19, 25, 44]. The general steps for a stochastic finite element to be used together with a FORM reliability method are as listed below:

1. The finite element formulation has to be performed. The element stiffness matrices should be formed. The global stiffness matrix for the system has to be assembled and the nodal force vector created.

2. The value of the displacements has to be found by solving Equation 4.12 for d.

3. In this step the desired response of the system should be calculated for the displacements obtained in the previous step. If the strength limit state is being evaluated, then the desired response is the truss element stresses. The desired response is computed by transforming the nodal displacement using Equation 4.13 below:

$$E = Q^T d + E_0 \tag{4.13}$$

   In Equation 4.13, E is the desired response of the system, and E_0 is the system response for $d = 0$ (for instance, initial stress in the elements). If in Equation 4.13, the desired response is the internal stress in elements, then attention is drawn to Equation 4.11 as shown below:

$$\sigma = E$$
$$Q = \frac{E_M}{L_e}\begin{bmatrix}-c & -s & c & s\end{bmatrix}$$

   In fact, Equation 4.11 is a special case of Equation 4.13 where $E_0 = 0$ since in the matrix formulation it is assumed that there is no initial stress in the elements. It should also be mentioned that the modulus of elasticity of the material is shown with E_M to avoid confusion of E which represents the response of the system (Load effect). If the element internal force is to be obtained Q is expressed as in Equation 4.14:

$$Q = \frac{E_M A_e}{L_e}\begin{bmatrix}-c & -s & c & s\end{bmatrix} \tag{4.14}$$

4. Once the response of the system is obtained, the corresponding value of the performance

function can be computed:

$$G(X) = \{R(X), E(X)\} \tag{4.15}$$

In Equation 4.15, R is the vector of element resistances and E is the response vector or the vector that contains element internal stresses (forces). X is identified as the vector of random variables in the original space. As a result, for a truss structure with n elements:

$$R = \begin{bmatrix} R_1(X) \\ R_2(X) \\ \vdots \\ R_n(X) \end{bmatrix} \text{ and } E = \begin{bmatrix} E_1(X) \\ E_2(X) \\ \vdots \\ E_n(X) \end{bmatrix}$$

Where R_n and E_n are the resistance and the internal stress in each bar with respect to the input random variables.

5. The random variables need to be in the normal Gaussian space. This can be done by using a transformation matrix. Let's assume that two random variables exist in the SFEM reliability evaluation problem such as X_1 and X_2. To transform these variables into Gaussian normal variables Equation 2.8 can be used:

$$Y_1 = \frac{X_1 - \mu_{X_1}^N}{\sigma_{X_1}^N} \text{ and } Y_2 = \frac{X_2 - \mu_{X_2}^N}{\sigma_{X_2}^N}$$

This transformation can be done in matrix form for these variables:

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sigma_{X_1}^N} & 0 \\ 0 & \frac{1}{\sigma_{X_2}^N} \end{bmatrix} \times \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} -\frac{\mu_{X_1}^N}{\sigma_{X_1}^N} \\ -\frac{\mu_{X_2}^N}{\sigma_{X_2}^N} \end{bmatrix} \tag{4.16}$$

Or in mathematical form:

$$Y = BX + A \tag{4.17}$$

Equation 4.16 can be generalised for n number of variables with the same formulation. As a result, it is possible to transform the random variables into the equivalent standard normal space.

6. The calculation of the derivatives of the limit state function in the equivalent standard normal space is the only necessary step left to provide all the "essentials" for the Newton-Raphson FORM reliability analysis. The matrix formulation for this is based on the classical perturbation sensitivity analysis or the chain rule of differentiation. The formulation can be illustrated for the two random variables case and then be expressed generally. If the limit state function is expressed with respect to two random variables X_1 and X_2:

$$G = G(X_1, X_2)$$

In order to get the derivatives of G in terms of transformed variables Y_1 and Y_2 by exploiting the chain rule of differentiation:

$$\frac{\partial G}{\partial Y_1} = \frac{\partial G}{\partial R} \times \frac{\partial R}{\partial X_1} \times \frac{\partial X_1}{\partial Y_1} + \frac{\partial G}{\partial E} \times \frac{\partial E}{\partial X_1} \times \frac{\partial X_2}{\partial Y_2}$$

$$\frac{\partial G}{\partial Y_2} = \frac{\partial G}{\partial R} \times \frac{\partial R}{\partial X_2} \times \frac{\partial X_2}{\partial Y_2} + \frac{\partial G}{\partial E} \times \frac{\partial E}{\partial X_2} \times \frac{\partial X_2}{\partial Y_2}$$

The values of $\frac{\partial X_1}{\partial Y_1}$ and $\frac{\partial X_2}{\partial Y_2}$ are obtained from the transformation equation (Equation 2.8), and it equates the equivalent normal standard deviation of each random variable, so the derivatives can be shown as below:

$$\frac{\partial G}{\partial Y_1} = \frac{\partial G}{\partial R} \times \frac{\partial R}{\partial X_1} \times \sigma_{X_1}^N + \frac{\partial G}{\partial E} \times \frac{\partial E}{\partial X_1} \times \sigma_{X_1}^N$$

$$\frac{\partial G}{\partial Y_1} = \frac{\partial G}{\partial R} \times \frac{\partial R}{\partial X_1} \times \sigma_{X_1}^N + \frac{\partial G}{\partial E} \times \frac{\partial E}{\partial X_1} \times \sigma_{X_1}^N$$

It is clear that to get a matrix form for the derivative of the limit state function with respect to equivalent standard normal variables, one can write:

$$\nabla G(Y) = B^{-1} \times \nabla G_x(R, E) \tag{4.18}$$

In Equation 4.18 B^{-1} is the inverse of B from Equation 4.17 and $\nabla G_x(R, E)$ is the gradient of G in terms of $X, R,$ and E. If Jacobians of the limit state function are formed, then Equation 4.18 can be written as shown in Equation 4.19:

$$\nabla G(Y) = B^{-1} \times J^T \times \nabla G(R, E) \tag{4.19}$$

In Equation 4.19 J is the Jacobian matrix. Its dimension is $2 \times n$ where n is the number of random variables. The first row of Jacobian matrix is $\frac{\partial R}{\partial X_j}$ and the second row is $\frac{\partial E}{\partial X_j}$. J^T is the transpose of the Jacobian matrix. For a two random variable case, Equation 4.18 is demonstrated as below:

$$\begin{bmatrix} \frac{\partial G}{\partial Y_1} \\ \frac{\partial G}{\partial Y_2} \end{bmatrix} = \begin{bmatrix} \sigma_{X_1}^N & 0 \\ 0 & \sigma_{X_2}^N \end{bmatrix} \times \begin{bmatrix} \frac{\partial R}{\partial X_1} & \frac{\partial E}{\partial X_1} \\ \frac{\partial R}{\partial X_2} & \frac{\partial E}{\partial X_2} \end{bmatrix} \times \begin{bmatrix} \frac{\partial G}{\partial R} \\ \frac{\partial G}{\partial E} \end{bmatrix}$$

It is usually easy to calculate the derivative of R in terms of random variables. However, the computation of the derivative of E can be difficult. Nevertheless, it can be calculated taking the derivative of Equation 4.13 with respect to X_j:

$$\frac{\partial E}{\partial X_j} = \frac{\partial Q^T}{\partial X_j} d + Q^T \frac{\partial d}{\partial X_j} + \frac{\partial E_0}{\partial X_j} \tag{4.20}$$

In order to calculate the derivative of displacement with respect to the random variable X_j, Equation 4.12 is used:

$$d = K^{-1} \times F \tag{4.21}$$

Then, the derivative of Equation 4.21 becomes:

$$\frac{\partial d}{\partial X_j} = K^{-1}\frac{\partial F}{\partial X_j} + \frac{\partial K^{-1}}{\partial X_j}F \tag{4.22}$$

In order to shed the term $\frac{\partial K^{-1}}{\partial X_j}$ the derivative of Equation 4.23 below can be used:

$$K \times K^{-1} = 1 \tag{4.23}$$

Therefore, the derivative of Equation 4.23 with respect to X_j becomes:

$$\frac{\partial K^{-1}}{\partial X_j} = -K^{-1}\frac{\partial K}{\partial X_j}K^{-1} \tag{4.24}$$

Replacing Equation 4.24 to 4.22 yields:

$$\frac{\partial d}{\partial X_j} = K^{-1}\frac{\partial F}{\partial X_j} - K^{-1}\frac{\partial K}{\partial X_j}d \tag{4.25}$$

Using Equations 4.20 and 4.25, the value of $\frac{\partial E}{\partial X_j}$ can be obtained. It is recommended that the force vector, stiffness matrix, displacement vector, etc be calculated together with their derivatives in the finite element formulation. All these calculations are highly dependent on the random variables considered. For instance, for truss structures the stiffness matrix is only in terms of modulus of elasticity, element area, and element geometry, so if all of these variables are considered as deterministic, then all of the derivatives of the stiffness matrix in Equations 4.20 and 4.25 are zero.

The partial derivatives of the stiffness matrix with respect to random variables can be obtained if the element area and modulus of elasticity are assumed to be random variables while the geometrical parameters of the stiffness matrix such as element length and element orientation angle are considered to be deterministic.

Element stiffness matrix of Equation 4.9 (planar truss) can be shown as below as well:

$$K_e = \begin{bmatrix} a & c & -a & -c \\ c & b & -c & -b \\ a & -c & a & c \\ -c & -b & c & b \end{bmatrix} \tag{4.26}$$

In Equation 4.26 the elements of the matrix are:

$$a = \frac{EA}{L_e} \times \cos^2\theta \tag{4.27}$$

$$b = \frac{EA}{L_e} \times \sin^2\theta \tag{4.28}$$

$$c = \frac{EA}{L_e} \times \sin\theta \times \cos\theta \tag{4.29}$$

To get the derivatives of the stiffness matrix, partial derivatives of the stiffness matrix components a, b, and c could be obtained; therefore, For the partial derivatives with respect to modulus of elasticity $\frac{\partial K}{\partial E}$:

$$a = \frac{A}{L_e} \cos^2\theta \tag{4.30}$$

$$b = \frac{A}{L_e} \sin^2\theta \tag{4.31}$$

$$c = \frac{A}{L_e} \sin\beta \cos\theta \tag{4.32}$$

For the partial derivative with respect to element cross sectional area $\frac{\partial K}{\partial A}$:

$$a = \frac{E}{L_e} \cos^2\theta \tag{4.33}$$

$$b = \frac{E}{L_e} \sin^2\theta \tag{4.34}$$

$$c = \frac{E}{L_e} \sin\theta \cos\theta \tag{4.35}$$

Using Equations 4.30 to 4.35, partial derivatives of the element stiffness matrix can be readily formed and used in FORM reliability calculations.

4.3.4.4 Stochastic Finite Element Method (SFEM) using the finite difference approach

In Section 4.3.4.3 the necessary matrix formulations for a SFEM method that is based on classical perturbation sensitivity method were shown. However, it is possible to avoid all of the above-mentioned matrix formulations by applying the finite difference approach. In this case all the derivatives of the performance function are calculated using the steps mentioned in Section 4.3.3.1 for the finite difference approach. Firstly, finite element analysis is run to calculate the value of limit state function at the initial design point values. Secondly, the finite element analysis is performed while the values of the input random variables are perturbed one by one while keeping other input random variables at their initial values. Finally, the third step of the finite difference approach is used to calculate the derivatives. The results of this derivation can be used in the FORM method described in Chapter 3 which uses the Newton-Raphson recursive procedure. In the mentioned FORM method only the values and the gradients of the performance function need be computed at each iteration point. Therefore, it is possible to calculate the value of the limit state function at each iteration point by simply performing a deterministic finite element analysis of the structure, and then by using the finite difference method the derivatives can be computed with respect to the random variables as mentioned above.

When the finite difference method is used to calculate the derivatives of the limit state function $g(X_1, X_2, ..., X_n)$ in a FORM reliability analysis, it is needed to do $n+1$ finite element analyses at each iteration. One is needed to calculate the value of limit state function at the initial point and other finite element analyses to calculate the approximate derivatives with respect to each of the random variables of the limit state function. If it is assumed that there are (m) required iterations for the FORM method to converge, then there will be $m(n+1)$ deterministic finite element analyses for the whole procedure in order to get the reliability index. For large structures, it can seem time-consuming; however, in practice usually commercially available finite element packages are used and access to the source code is not provided. Even if the access to the source code were provided, the programming could be time-consuming and exhausting per se. In the following chapter, it is shown that it is possible to use a commercially available finite element package like Srand7 version 2.4.2 to perform the reliability analysis with the help of its application programming interface (API). In summary, performing a stochastic finite element analysis using finite difference method could provide convincing outcomes if it is used with care and small enough perturbation sizes are selected, not to mention its practicality [19]. This method is implemented in Chapter 5 where a step-wise procedure is shown for this method.

4.4 Conclusion regarding reliability analysis methods for implicit limit state functions

Different methods of coping with implicit limit state functions and the concept of SFEM were illustrated above. All of the aforementioned methods have their own advantages and disadvantages. The response surface method can be a useful approach, but when the reliability of larger structures is investigated the number of random variables increases, and if a method such as the factorial method is used the number of deterministic finite element analyses that has to be done will increase exponentially, and it can be time-consuming. Even if the number of random variables is not that large, the calculation is only as accurate as the approximated closed-form performance function, and in case of highly non-linear performance functions this method cannot be rigorous enough. On the other hand, Monte Carlo simulation needs a large number of finite element analyses which can be very time-consuming especially for the cases of low probability of failure since the number of simulations has to be increased. However, sensitivity-based approaches, specifically the finite difference method, are generally more efficient and elegant than the Monte Carlo simulation and the response surface method [19]. Particularly, when a commercial finite element software package is used the finite difference method can be quite straightforward to use. Classical perturbation needs alterations in the finite element code, and it may need a considerable amount of programming. However, if the intention is to provide an integrated software environment for the purpose of stochastic finite element calculation, the classical perturbation method can be a useful sensitivity-based method. Iterative perturbation is only applicable when a non-linear finite element analysis is intended.

It should be noted that all of the aforementioned methods are implemented in Chapter 5 to investigate their practicality and evaluate their efficiency to perform a component level reliability analysis for truss structures.

Chapter 5

Component Level Reliability Analysis of Trusses

5.1 Introduction

In Chapter 4 different methods of performing a component level structural reliability analysis considering the applicability of implicit limit state functions were explained. In this chapter the focus is on the development of computer algorithms for a component level reliability analysis of indeterminate truss structures. Throughout the chapter, computer algorithms and programs are developed for the reliability analysis where an investigation is performed on the practicality and efficiency of the methods explained in Chapter 4. In Section 5.2 through 5.4, the use of a commercial finite element package is investigated on a simple statically determinate structure where due to the determinacy of the structure the obtained results are easier to check. This will serve the purpose of establishing a robust algorithm where the post-processing of the results taken from a commercial finite element package is performed through an API link to the programming language. In other terms, the focus in this section is more on developing an algorithm for the Newton-Raphson FORM method using a finite difference sensitivity-based method through the implementation of a commercial finite element package. In Section 5.5 different methods including SFEM, Response surface, and Monte Carlo simulation methods are employed for a computerised component level reliability analysis of a simple indeterminate truss structure. In this section, developed algorithms for these methodologies are used to develop computer programs for the reliability analysis. Finally, the methodologies are discussed thoroughly. In short, the purpose of this chapter is to provide a comprehensive set of computerised methods on component level reliability analysis of truss structures.

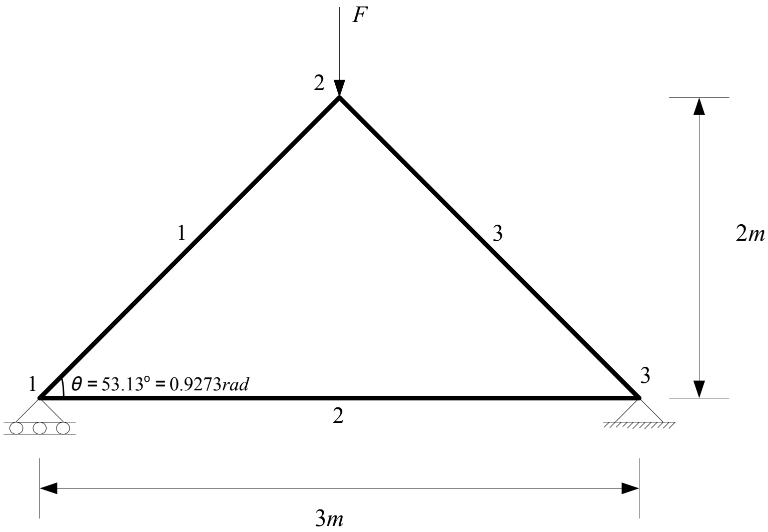

Figure 5.1: A three-bar truss model

5.2 Reliability analysis of a statically determinate truss structure

A statically determinate three-bar truss structure is considered to develop the methodology for the FORM reliability analysis method. Owing to the simplicity of the structure the results of the FORM approach, that is based on the concept of finite difference sensitivity-based method which uses the Newton-Raphson recursive procedure, can be readily verified. The truss model is shown in Figure 5.1.

As it can be seen from the figure, the model is a statically determinate truss structure with three elements. It is possible to calculate the forces in every member explicitly in terms of the applied force F. However, it is assumed that the forces in the members are not explicitly available with respect to the applied force. In other terms, it is assumed that the strength limit states for the members of the truss are implicitly in terms of the applied load. However, the explicit relationship will then be used to confirm the obtained results.

A model of the structure is created in the commercial finite element software Strand7 to be used with the finite difference approach (Figure 5.2). Using the application programming interface (API) of the finite element program [1], it will possible to perform the FORM reliability analysis. The details are discussed in Section 5.3 as follows.

5.3 SFEM algorithm based on finite difference approach

In order to compute the reliability of each element of the truss, an algorithm that uses a combination of the FORM method based on Newton-Raphson recursive process and the finite element application programming interface needs to be developed. The steps of the algorithm to be used

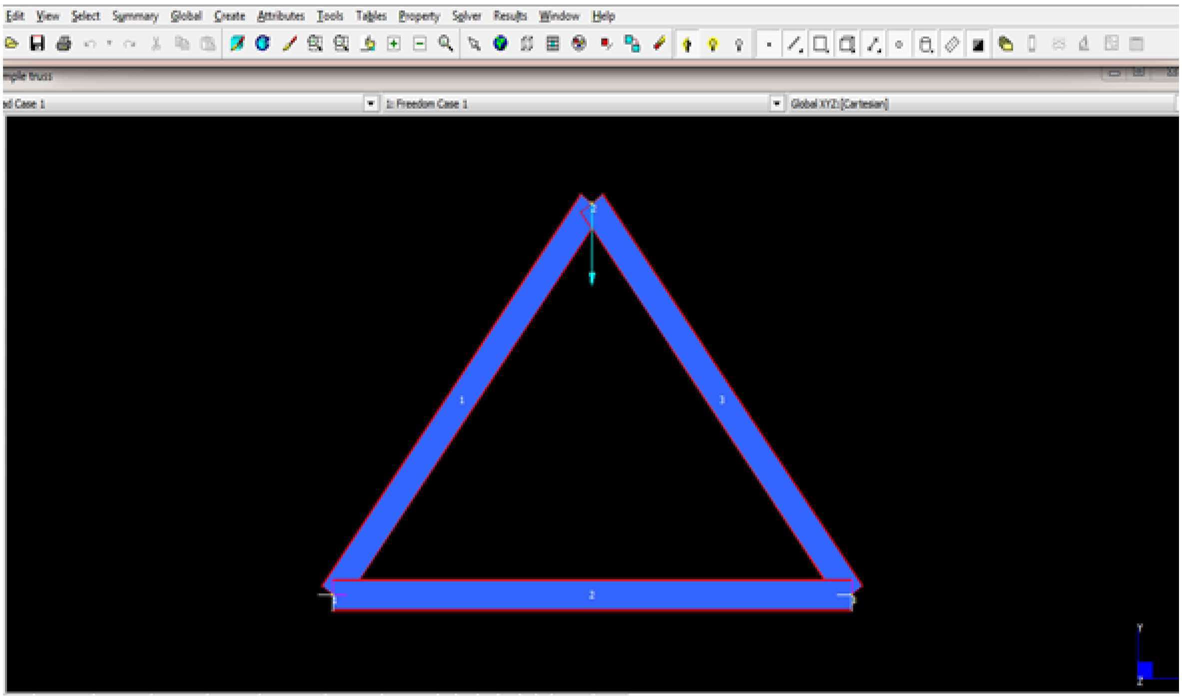

Figure 5.2: Finite element model of the 3-bar truss in Strand7

for the component reliability calculation are as follows. The algorithm is developed using the concepts discussed in Section 4.3.4.4 of Chapter 4.

1. Open the Strand7 finite element model (in this case the truss model).
2. Enter the stochastic parameters of the applied force and guess a point as the initial design point to start the FORM analysis in the developed reliability analysis program.
3. Run the finite element analysis (run the linear static solver of Strand7, and open the linear static result file (file with .LSA extension)).
4. Extract the obtained results to calculate the value of the performance function for each of the components. Store these values in a vector called G1.
5. Calculate the equivalent normal standard parameters for non-normal random variables.
6. Transform the random variables into standard normal random variables using Equation 2.8.
7. Start calculating the derivatives of the limit state function by perturbing the variables by a small amount one at a time.
8. Run the linear static solver of Stand7 again, and open the result file.
9. Extract the values for each component, and calculate the performance function for each component, and store the results in a vector called G2.

10. Calculate the derivatives using the following formula:

$$\Delta G = \frac{G2 - G1}{\Delta X} \tag{5.1}$$

11. Repeat steps 8 to 12 to obtain the derivative for each random variable.

12. Compute the derivatives in the normal Gaussian space through Equation 3.14.

13. Get the new design point using Equation 3.16.

14. Acquire the reliability value using Equation 3.17.

15. Calculate the coordinates of the new design point in the normal space via Equation 3.18.

16. Using the new design point values run the finite element analysis. And extract the results from the results file.

17. Calculate the value of the performance function for each element and store it in a vector called $G3$.

18. Check for the convergence of $G3$ and β ($G3$ has to be close to zero and difference between two successive β values has to be smaller than a tolerance level). If convergence criterion is met finish the procedure, otherwise repeat the procedure.

The algorithm explained above is based on the assumption that the resistance is deterministic. However, very simply and with some little changes, it can be altered in a way that it can be used for cases where resistance is not deterministic since the limit state function is mostly explicit with respect to resistance variables. A flow diagram of the whole procedure is depicted in Figure 5.3. The results that were obtained from this algorithm were confirmed with Monte Carlo simulation and the Integration method.

5.4 Evaluation of the results of the 3-bar truss model

According to the algorithm explained above a finite element model was developed for the structure (Figure 5.2). In Appendix I of the thesis a complete account of using the application user interface of Strand7 along with how to access the API is provided [1]. The necessary Matlab codes for the Finite element analysis using the Strand7 API, performing the FORM reliability analysis, Monte Carlo simulation, and integration method are provided in Appendix II.

In this example, it is assumed that the applied load F follows a lognormal distribution with a mean value of 170 and a standard deviation of 34 (or a coefficient of variation of 0.2). From Figure 5.1, it is obvious that the internal forces of the truss members 1 and 3 are the same. Also, These two members are acting in compression while member 2 will be in tension. The deterministic resistances of the members are shown in the table below:

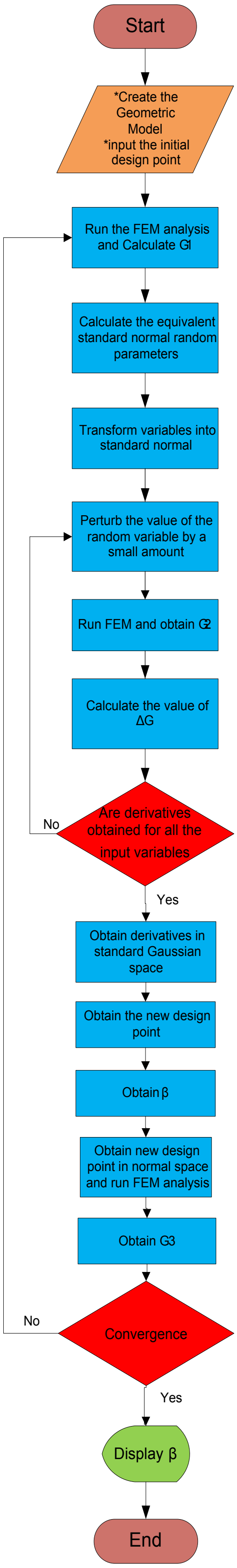

Figure 5.3: Flowchart of SFEM using Strand7 FEM package

Table 5.1: Resistance values for the simple three-bare truss example

Member Type (Tension/Compression)	Deterministic Resistance Value
Tension	130
Compression	190

The internal forces that are caused in members 1 and 3 from the equilibrium of node 1 can be obtained. The value of the internal force in those two members is equal to:

$$F_1 = F_3 = \frac{F}{2 \times \sin(\theta)} \tag{5.2}$$

The internal force in member 2 is obtained as below:

$$F_2 = \frac{F}{2 \times \tan(\theta)} \tag{5.3}$$

where $\theta = 0.9273$ rad.

Through Equations 5.2 and 5.3, it is possible to get a closed-form function in terms of the random variable F. These two equations are used in the Monte Carlo analysis and the integration method to confirm the obtained results from the algorithm explained in Section 5.3. It should be noted that it is possible to use the finite element model to calculate the component reliability of this simple truss with the Monte Carlo simulation method. However, here simply by using the two equations above Monte Carlo simulation can be done considerably faster. As a result, the finite element analysis was not used in Monte Carlo simulation.

5.4.1 Results using the SFEM algorithm

The algorithm that was developed in Section 5.3 is used to calculate the reliability index of the components of the truss structure shown in Figure 5.1. The finite element model that is shown in Figure 5.2 was used to perform the deterministic finite element analysis of the structure. The following results were obtained from the algorithmic analysis.

Table 5.2: Reliability index using FORM method

Component number	Reliability index
1	3.034
2	3.697
3	3.034

It should be noted that the whole procedure took 12 minutes and 43 seconds on an Intel Core i5 machine where the total number of iterations for the whole structure was 45. In general, the linking between the finite element software's API and the programming language was observed to take the most of the computational time. However, there are some other parameters involved in the computational time such as the convergence criterion, selection of initial design point values, and so on. It should also be mentioned that the computational time doesn't follow a linear relationship with the number of components (or how large a structure is). For instance, for the ten-bar truss structure considered in Section 5.5, it takes about 20 minutes for the algorithm to converge where the resistance is also a random variable.

5.4.2 Results of Monte Carlo simulation

A direct Monte Carlo analysis of the simple truss of Figure 5.1 was performed using 1,000,000 simulations. A full Matlab transcript of this method is presented in Appendix II. In the computer programming for the Monte Carlo simulation the concept of vectorisation was used in order to avoid the so-called "for loops" which makes the process more time-consuming. First, a vector of one million random variables in the interval [0, 1] was formed using Matlab's built-in function "rand". Secondly, using the methods explained in Chapter 3 random variables were generated for the applied force F which follows a lognormal distribution. All these values where substituted into Equations 5.2 and 5.3 in order to evaluate the strength limit state function of the members. For tension members the limit state function is of the form:

$$G = 130 - E_t \tag{5.4}$$

Where E_t is the internal tension force in the component (member 2). And the limit state function for compression has the form:

$$G = 190 - E_c \tag{5.5}$$

In Equation 5.5, E_c is the internal compression force in the component (members 1 and 3). From Direct Monte Carlo simulation the reliability indices were calculated for each component of the truss. These values are presented in Table 5.3 below.

Table 5.3: Reliability indices using DMCS

Component number	Reliability index
1	3.042
2	3.654
3	3.042

5.4.3 Results from the integration method

The integration method uses the concepts explained in section 3.4 of Chapter 3. The method is basically an approximate integration of the failure area. In other words, the area of the failure region will be determined by using a rough numerical integration method. It should

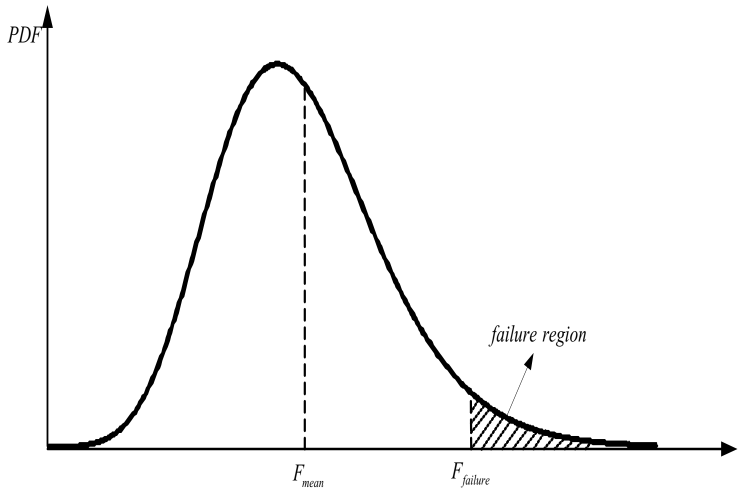

Figure 5.4: The Integration method for the 3-bar truss structure

be mentioned that this method can only be used for this simple example where the explicit relation between the internal force of each member and the applied force is clear. In the case of indeterminate structures where the resistances are also random variables this method is very hard to use unless a response surface method is applied. Thus, this method was only used for the confirmation of the results in this case where the truss structure is statically determinate.

It was shown that the deterministic resistance of members 1 and 2 in compression is 190, and for member 2 in tension, it is 130. Since the explicit relation between the member forces and the applied load is available, by substituting the resistances in Equations 5.2 and 5.3, it is possible to find the load that causes failure in each of those members. Knowing the failure load and distribution of the applied load, the failure probability can be found using the numerical integration expression demonstrated as Equation 3.42 in Chapter 3.

The failure load values for the compression members 3 and 1 as well as the tension member 2 are shown below.

From Equation 5.2 the failure load for the compression members 1 and 3 can be computed as:

$$F_{failure_{1\ \&\ 3}} = F_1 \times 2 \times \sin(0.9273) = 190 \times 2 \times \sin(0.9273) = 304.000$$

From Equation 5.3 the failure load for the tension member 2 is:

$$F_{failure} = F_2 \times 2 \times \tan(0.9273) = 130 \times 2 \times \tan(0.9273) = 346.655$$

The concept is also shown graphically in Figure 5.4 above.

Numerical integration will be executed in the interval $[F_{failure}, 3\sigma]$. This range will be divided into many smaller areas and sum of all of these small areas will yield the value of the failure probability of that component which is the hatched area shown in Figure 5.4. Then, by just using Equation 2.12 the reliability index can be computed. The results of the reliability calculation for the members are shown in Table 5.4. The integration increment was chosen as 0.01. The Maltab transcript of this method is also provided in Appendix II of the thesis.

Table 5.4: Reliability index using integration method

Component number	Reliability index
1	3.038
2	3.697
3	3.038

5.4.4 Conclusion and comparison of the obtained results

The reliability of the members of a simple truss under a stochastic load effect and deterministic resistances was computed using three different methods of reliability analysis. Firstly, a stochastic finite element analysis based on FORM was performed which didn't need the explicit relationship between the applied force and member internal forces. Next, the reliability analysis was done using direct Monte Carlo simulation (DMCS). The number of simulations was 1,000,000 and here because the structure was statically determinate the explicit relation between the applied loads and resistances was used. Finally, the integration method was used to get the member reliability indices by exploiting Equations 5.2 and 5.3 which provide an explicit relationship between the applied load and member internal forces. The comparison between the results is obtained in Table 5.5 below.

Table 5.5: Reliability indices obtained using different methods

Member	DMCS	Integration	SFEM algorithm
1	3.042	3.038	3.034
2	3.654	3.697	3.697
3	3.042	3.038	3.034

It can be seen from Table 5.5 that the reliability indices calculated using the SFEM algorithm are close to the ones obtained through other methods. In fact, the other two methods confirm the results of the SFEM algorithm. In conclusion, SFEM algorithm can be effectively used to calculate the reliability index for the components of a truss structure. In Section 5.4 a comprehensive investigation on different methods of component level reliability analysis for truss structures is performed using a 10-bar statically indeterminate structure where both resistance and load effect are random variables.

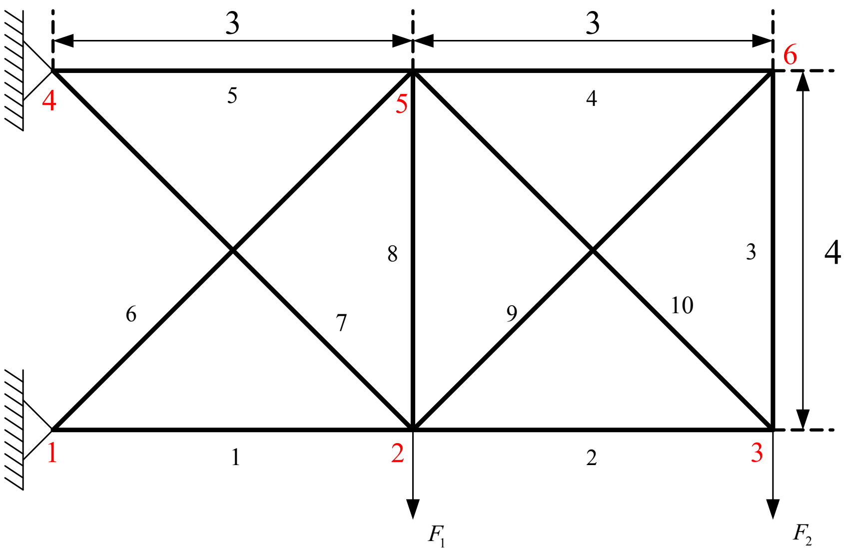

Figure 5.5: Schematic model of the 10-bar truss structure

5.5 Component level reliability investigation of a statically indeterminate 10-bar truss structure

In the previous section an algorithm for using a Newton-Raphson FORM reliability analysis based on a finite difference sensitivity method using commercial finite element software was established for a statically determinate 3-bar truss structure. In practice, however, some of the truss structures are statically indeterminate and have some degree of redundancy. Therefore, it is necessary to consider a statically indeterminate truss structure for the investigation of different reliability evaluation methods proposed in Chapter 4 as well as the algorithm that was developed in Section 5.3. This way, it is possible to assess the functionality and practicality of different methods for a computerised component level reliability analysis for the truss structures. For this purpose, a 10-bar truss structure was selected for the investigation. This type of truss structure is used in many documents of reliability literature and research ([2,48]). Nevertheless, this model was modified for the purpose of this thesis, and it doesn't have the same resistances, applied loads, and geometrical dimensions as the ones used in the aforementioned literature. The model is shown schematically in Figure 5.5.

In Figure 5.5, it is shown that the structure has the horizontal spans of 3m and its height is 4m. The node numbers are shown with red figures, and two forces are applied on nodes 2 and 3. These forces are called F_1 and F_2 respectively. F_1 and F_2 are both in terms of a random variable so that a constant load pattern is kept. In other terms, the truss components can only be in tension or compression since there is only one load effect variable. The resistance and load effect details of the structure are discussed in the sections that follow. It should also be mentioned that the structure is statically indeterminate to degree of 2 (or has two degrees of redundancy).

5.5.1 Resistances and Applied Loads of the Model

For the reliability analysis of the structure it is needed to define the model resistances and load effects. Since the model is an ideal truss structure, its members will either be in tension or compression. Therefore, proper members have to be considered for the different elements of the truss so that they are capable of resisting the member internal forces. The loads F_1 and F_2 are applied on nodes 2 and 3 where $F_1 = F$ and $F_2 = 0.8F$. These forces should be at such a level where the member capacities are equal to the internal member forces for the deterministic design (or at least they have low redundant capacity).

Table 5.6: Section properties of the ten-bar truss structure

Member	Length(m)	Section(Hollow)	Area(mm^2)	r(mm)
1	3	127×5	1916	43.2
2	3	101.6×3	929	34.9
3	4	32×2	188	10.6
4	3	32×2	188	10.6
5	3	101.6×4	1226	34.5
6	5	165.1×4.5	2270	56.8
7	5	114.3×3	1049	39.4
8	4	38×3	331	12.5
9	5	101.6×3	929	34.9
10	5	101.6×2.5	778	35.0

Accordingly, the forces will be: $F_1 = 290$ and $F_2 = 0.8 \times 290 = 232$. These forces cause members 2, 7, and 10 have low redundant capacities. However, it is not achievable for all the members since the structure is statically indeterminate (to the degree of 2), and the redistribution of forces takes place due to the redundancy of the structure when the member cross sections are changed. The fact that some members have low redundant capacity is useful in that a reasonable reliability index value can be expected for these members (a reliability index of around 2 or 3 according to the partial design factor proposed in South African codes).

As is shown in Table 5.6 above, the sections used for the truss structure are steel hollow sections. Also in other columns of the table the area and the radii of gyration of the sections used are given.

In Table 5.7 the member forces at characteristic values of $F_1 = 290$ and $F_2 = 232$ together with their resistances are given. The details of the resistance computations are given in Section 5.5.2 below.

5.5.2 Statistical properties of the resistance and load effect

In order to perform the reliability analysis for the components of the truss structure, the stochastic properties of the applied loads and resistances are essential. This can be achieved by

Table 5.7: Internal force and resistance of members

Member	Internal force(kN)	Type	Resistance
1	-379.65	Compression	387.500
2	-145.160	Compression	150.568
3	38.453	Tension	59.370
4	28.840	Tension	59.370
5	369.335	Tension	386.340
6	-325.558	Compression	356.083
7	326.942	Tension	330.428
8	66.889	Tension	104.205
9	-48.066	Compression	71.452
10	241.934	Tension	245.174

Table 5.8: Stochastic properties of the load effect and resistance

Parameter	PDF	Mean	Standard deviation
F	Extreme type I	174	60.9
f_y	Lognormal	407	28.5

transforming the characteristic values of the load effect and resistance to a mean and standard deviation for a proposed probability distribution. It is recommended by Holicky [22] to use the 60% of the characteristic value of applied load as the mean value and 35% of the mean value as the standard deviation. it is also suggested that an Extreme Type I (Gumbel) distribution be used for the load effect if a time-invariant reliability analysis is desired [22]. As a result, the applied load will have a mean value of 174 kN and standard deviation of 60.9 kN.

For the resistance parameters, it is observed in the literature that the only resistance parameter that has a great effect on the reliability of steel structures is the yield stress. Other resistance parameters such as modulus of elasticity (E) and area (A) do not affect the reliability significantly [28], and hence are not considered here as random variables. This will make the reliability formulation less complex and accordingly decreases the computational time. For the yield stress (f_y) literature suggests a lognormal distribution [17, 22, 28].

In order to compute the mean value of the yield stress the following expression is used [22]:

$$\mu_X = \frac{X_k}{1 - kw_X} \tag{5.6}$$

In Equation 5.6, the value of k is recommended to be taken as 2 [22]. Once the mean value is available from Equation 5.6, the standard deviation can be taken as 7% of the mean value. This will lead to the mean value of 407 MPa and standard deviation of 28.5 MPa . The load resistance parameters are outlined in Table 5.8 above.

In Table 5.8 the type of random variables (parameters) are shown in the first column. In the second column the type of probability density function (pdf) of the random variables is shown, and in the last two columns the mean values and standard deviations of the parameters are shown respectively.

Considering all of the assumptions mentioned above for the probability distribution functions of load effect and resistance, a case where both the load effect and the resistance follow a normal distribution will also be considered. The reliability evaluation results will be obtained for both cases. Later, a comparison is made between the obtained component reliability index results for these two cases.

5.5.3 Limit State Equations for the Structural Components

In the reliability analysis of the structure, the strength limit state is considered. The structure is a truss structure with pinned connections. Accordingly, the members of the structure can be either in tension or compression. This implies that for compression members the buckling capacity of the members should be considered and for tension members the yielding capacity of the gross section needs to be considered. In general, it is supposed that the truss connections will not fail, and for tension members it is assumed that failure mode of the fracture of the net section will not happen. Overall, a glance at the resistance elements of the limit state function shows that the difference between compression and tension members is only by a factor χ which further reduces the capacity of the compression members in order to include the buckling effect. The two limit states are shown Table 5.9 below.

Table 5.9: Limit state functions for the ten-bar truss structure

Member type	Strength performance function
Tension	$G = A \times f_y - E(F)$
Compression	$G = A \times f_y \times \chi - E(F)$

In order to calculate χ Equations 5.7 and 5.8 can be used [4, 29]:

$$\lambda = \frac{k.L}{r}\sqrt{\frac{f_y}{\pi^2 E}} \tag{5.7}$$

$$\chi = \left(1 + \lambda^{2n}\right)^{-1/n} \tag{5.8}$$

In Equation 5.8 (n) can have a value of 1.34 or 2.24. Since hollow structural steel sections are assumed to meet the manufacturing criteria of SANS 657-1, the value of 1.34 is used. It should be noted that the buckling reduction factor χ is considered as a deterministic variable and the specified value of f_y is used to calculate it. In other terms, the yield stress in χ is treated as if it were a deterministic value. The reason for this is to simplify the reliability evaluation. In fact, the effect of considering f_y as a deterministic value in χ is not considerable so that it can be ignored in order to make the reliability calculations simpler. This will be demonstrated in

Section 5.5.6.3.

Another component of the performance function is the load effect. Here, the load effect is the stress in the member which is in terms of the applied load F on the truss structure (applied on nodes 2 and 3). However, due to the indeterminacy of the truss a closed-form expression in terms of F will not be available for members unless a response surface method (regression in other words) is used. As a result, the limit states are implicit with respect to the load variables, and reliability analysis methods mentioned in Chapter 4 should be used to find the reliability index value of each member.

5.5.4 Methods used for performing reliability analysis

The proposed methods in Chapter 4 are used to evaluate the reliability of the statically indeterminate structure. As it was mentioned in Section 5.5.3, a closed-form or explicit relation between the member forces and the applied load (F) is not available due to the indeterminacy of the structure; therefore, it is not possible to obtain the first derivative of the limit state with respect to the applied forces by simply differentiating the limit state function if the FORM method is used. In general, the following methodologies can be used for a component level reliability analysis of the structure as discussed in Chapter 4:

- First Order Reliability Methods (FORM)
  - Newton-Raphson recursive procedure based on finite difference method
  - A fully stochastic finite element formulation based on classic perturbation
- Simulation-based methods
  - Direct Monte Carlo simulation
  - Latin Hypercube sampling Monte Carlo
  - Updated Latin Hypercube sampling Monte Carlo
- Response surface method
  - Response surface method based on *factorial* experimental design

All of the methods mentioned above are used for the reliability analysis of the structure shown in Figure 5.5. Firstly, the reliability analysis is performed using the methodology developed in Section 5.3. Next, a fully stochastic finite element formulation is developed for the reliability analysis.

The response surface method is used to find a closed-form expression for the members limit state functions, and simulation methods are used for the evaluation of the obtained limit states. This is achieved by performing a regression analysis such as the least square method to find the approximate closed-from expression between the applied force and internal member forces caused by F_1 and F_2 which is equal to $0.8 \times F_1$. This method can be very efficient since the analysis is linear static and a linear relationship between the external force and internal member forces exists. In fact, for the structure within the linear elastic range one can readily find the relationship between the applied load and member forces by performing the finite element analysis for a specified value of the applied force F, and then by simply dividing the applied force by the internal force of each member get a coefficient for each member. As a result, internal force in each member can be shown as in Equation 5.9:

$$f_i = m_i \times F \tag{5.9}$$

Where f_i is the internal force in each member, m_i is the factor that relates the applied force to internal force in each member.

Nevertheless, a regression-based response surface method will be used for the sake of showing the feasibility of the method and its application in a computerised reliability evaluation of the structure.

Simulation methods are another alternative way to do the component level reliability analysis of the whole structure. However, this method is not in fact a viable option if the application programming interface of Strand7 is used. This is due to the time-consuming process of linking between the finite element software and the programming language, and if a proper precision is intended, it can take days to complete. Accordingly, a finite element code for the structure is developed to make the direct Monte Carlo simulation of the whole structure a feasible option [15]. Moreover, different types of sampling methods mentioned in Chapter 3 will be implemented in the Monte Carlo simulation to investigate their efficiency as well.

These methods are all shown in detail in the following sections. All of the Matlab programming codes for each method are available in Appendix II.

5.5.5 Reliability analysis and the obtained results

In this section computerised analyses of the truss structure shown in Figure 5.5 are performed using the aforementioned methodologies. At the end of each section tabulated results of the reliability analysis are presented. First, the results from Newton-Raphson FORM reliability analysis which uses finite difference sensitivity method will be presented. Secondly, the results of a fully stochastic finite element analysis are given. Next, the results of a response surface method are shown, and, Finally, the obtained results form different simulation methods are presented. It should also be mentioned that for each method two case are considered: the case where load effect random variable is Extreme Type I (EV) while the resistance random variable is lognormal (Case I) and a case where both of the basic random variables are normal (Case II).

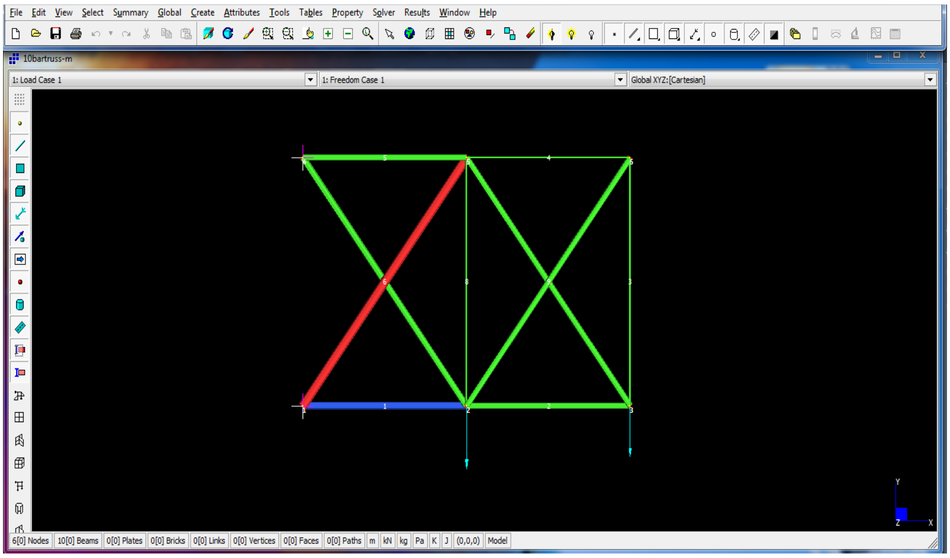

Figure 5.6: Finite element model of the 10-bar truss in Strand7

5.5.5.1 Finite Difference-based FORM reliability analysis

The proposed algorithm in Section 5.3 was implemented here to get the reliability of each truss member. Matlab transcript for the FORM method is provided in Appendix II of the thesis. The deterministic finite element analysis was done using Strand7 version 2.4.2 finite element package. The FEM model of the structure is shown in Figure 5.6. The application user interface (API) is used to connect the finite element software to the FORM reliability analysis program and to “feed” the finite element analysis results into the algorithm.

The total number of the iterations to do the reliability evaluation of the whole structure is 51, and it takes approximately 20 minutes for the evaluation process to complete for case I. For Case II, the total number of iterations is 20 and the process takes about 9 minutes for convergence. The computations are performed on an Intel Core i5 machine. The results are given in Table 5.10 and Table 5.11 below.

Table 5.10: Reliability indices and probabilities of failure (FORM-Case I)

Member	β	P_f
1	2.4850	0.00648
2	2.4593	0.00696
3	3.6274	1.43E-04
4	4.5379	2.84E-06
5	2.4852	0.00647
6	2.6263	0.00431
7	2.3855	0.00853
8	3.6502	1.31E-04
9	3.4987	1.34E-04
10	2.3944	0.00832

Table 5.11: Reliability indices and probabilities of failure (FORM-Case II)

Member	β	P_f
1	3.2634	0.00055
2	3.2157	0.00065
3	5.5295	1.61E-08
4	7.3373	1.09E-13
5	3.2639	0.00055
6	3.5298	0.00021
7	3.0797	0.00104
8	5.5759	1.23E-08
9	5.2670	6.93E-08
10	3.0960	0.00098

5.5.5.2 Fully SFEM-based FORM reliability analysis

A complete stochastic finite element formulation can also be used for the reliability analysis of the structural components. This means the calculation of the derivatives is performed by altering the finite element formulation of the structure. The programming code for this method is presented in Appendix II.

The algorithm that can be used for the indeterminate structure shown in Figure 5.5 is presented below.

1. Perform a FEM analysis of the structure based on the arbitrarily chosen initial design points as below.
   (a) Assemble the global force (F) and stiffness matrix for the structure.
   (b) Solve the equation $F = K.d$ for the displacements (d).
   (c) Calculate the vector of response for the internal member forces based on local coordinate of the element.

$$f = Q^T.d = \frac{EA}{L}\begin{bmatrix} -c & -s & c & s \end{bmatrix} d \tag{5.10}$$

2. Compute the value of g based on the results obtained in the previous step.

$$G(X) = \{R(X), E(X)\} \tag{5.11}$$

3. Get the equivalent normal parameters of the non-normal random variables. (this step is not required for normal random variables).
4. Calculate the derivatives of G with respect to the basic input random variables. The derivative of G is easily available in terms of resistance. The derivative with respect to the load effect is obtained as below.
   (a) Form the global stiffness matrix.

(b) Form the global vector of force derivative(s). Clearly, the derivative of the applied force with respect to the load effect is 1. Therefore, the vector of force derivative is readily formed by replacing F with 1. For the structure of Figure 5.5, the global force vector and its derivative vector are as below.

$$F = \begin{bmatrix} 0 & 0 & 0 & F & 0 & 0.8F & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\frac{\partial F}{\partial F} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0.8 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(c) Obtain the value of $\frac{\partial d}{\partial F} = K^{-1}F_I$ where F_I is the vector of global force derivatives, K^{-1} is the inverse of the global stiffness matrix.

(d) Compute the value of $\frac{\partial E}{\partial F} = \frac{EA}{L}\begin{bmatrix} -c & -s & c & s \end{bmatrix} \times \frac{\partial d}{\partial F}$, where $\frac{\partial d}{\partial F}$ is a 1-by -4 vector corresponding to the element dof's.

5. Calculate the reliability index of the element by:

   (a) Calculating the new design point using Equation 3.16.

   (b) Calculate β value via Equation 3.17.

6. Get the value of new iteration point in the original variable space using Equation 3.18.

7. Check the convergence of the algorithm.

8. Repeat the process for all of the components of the structure.

The results that were obtained using this method are presented in the tables 5.12 and 5.13 below.

Table 5.12: Reliability indices and probabilities of failure (SFEM-FORM-Case I)

Member	β	P_f
1	2.4506	0.00713
2	2.4291	0.00757
3	3.4845	2.47E-04
4	4.1902	1.39E-05
5	2.4546	0.00705
6	2.5900	0.00480
7	2.3541	0.00928
8	3.5189	2.17E-04
9	3.3743	3.70E-04
10	2.3671	0.00896

5.5.5.3 Response Surface Method (RSM)

The response surface method proposed in Chapter 4 is used to evaluate the reliability of the structure. For the ten-bar truss structure shown in Figure 5.5, the response surface approach

Table 5.13: Reliability indices and probabilities of failure (SFEM-FORM-Case II)

Member	β	P_f
1	3.2598	0.00056
2	3.2177	0.00065
3	5.5177	1.72E-08
4	7.3255	1.19E-13
5	3.2675	0.00054
6	3.5368	0.00020
7	3.0731	0.00106
8	5.6005	1.07E-08
9	5.2553	7.39E-08
10	3.0980	0.00097

can be performed in two ways. One approach is to find a closed-form (explicit) relationship between the applied load and the internal member forces. It will lead to finding a relationship such as $f_i = m_i \times F$ for each one of the truss members. A regression method is used to find the relationship between the applied force (F) and member internal forces (f_i). This approach can be referred to as a quasi-response surface method since it only deals with the response of load effect part of the performance function (only considers the load effect random variables). Another approach is to get the response of the whole structure for the performance function. This way a response surface can be formed with respect to all of the random variables involved in the limit state function (F and f_y). In such a way in each deterministic analysis the whole limit state function G needs to be assessed. Here, the first method was chosen since the resistances are already explicitly available with respect to the yield stress, and regression is only executed to approximate the response of the system due to the load effect random variable F.

A least-squares regression method is used to find a closed-form expression for the element internal forces with respect to the applied load. Five regression values were considered for the applied load F. These values are $\mu-2\times\sigma, \mu-\sigma, \mu, \mu+\sigma, \mu+2\times\sigma$ (factorial method). They are respectively 52.5, 113.1, 174, 234.9, 295.8. For each one of these values a deterministic finite element analysis is done, and each time the member internal forces are stored in a vector. Through all of the obtained values a regression is done and the response of each member to the applied load is evaluated. The obtained results from the finite element analysis for different members of the ten-bar truss are listed in Table 5.14.

Table 5.14: Internal forces due to five different applied loads

F	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
-52.2	-66.64	-26.13	6.92	5.19	66.47	-58.58	58.87	12.03	-8.65	43.55
-113.1	-144.39	-56.61	15.00	11.25	144.01	-126.92	127.56	26.06	-18.75	94.35
-174	-222.14	-87.09	23.08	17.31	221.56	-195.26	196.24	40.09	-28.85	145.15
-234.9	-299.89	-117.57	31.16	23.37	299.10	-263.60	264.92	54.12	-38.94	195.96
-295.8	-377.64	-148.06	39.23	29.42	376.65	-331.94	333.61	68.15	-49.04	246.76

The first Column of Table 5.14 shows the values of the applied loads and the columns 2 through 11 display the internal member forces in each of the components of the truss. The minus sign shows that the member is in compression, whereas the positive sign shows that the member is in tension. The minus sign for the applied loads is only to show the direction of the applied load, which means that the load is applied downwards. Using these values, it is possible to perform a least squares regression (using built-in functions of Matlab) and find an explicit expression for the truss members. Table 5.15 shows the obtained closed-form expression for the load effect of each member together with the resistances.

Table 5.15: Closed-form expressions for load effect and resistance

Component	Load Effect	Resistance
Member 1	$E_1 = -1.2767F$	$R_1 = 1.2293f_y$
Member 2	$E_2 = -0.5005F$	$R_2 = 0.4778f_y$
Member 3	$E_3 = +0.1326F$	$R_3 = 0.1885f_y$
Member 4	$E_4 = +0.0995F$	$R_4 = 0.1885f_y$
Member 5	$E_5 = +1.2733F$	$R_5 = 1.2260f_y$
Member 6	$E_6 = -1.1222F$	$R_6 = 1.1352f_y$
Member 7	$E_7 = +1.1278F$	$R_7 = 1.0490f_y$
Member 8	$E_8 = +0.2304F$	$R_8 = 0.3310f_y$
Member 9	$E_9 = -0.1658F$	$R_9 = 0.2257f_y$
Member 10	$E_{10} = +0.8342F$	$R_{10} = 0.7783f_y$

Looking at the expressions in the table, it can be seen that the intercept for all of the internal force explicit expressions is zero and the internal force in each member is just a coefficient multiplied by the applied external force (m). For the resistance, in the case of compression members the coefficient that reduces the resistance (χ) was multiplied by the area and was divided by 1000 so that the resistance values are in kN. In case of tension members the coefficient is just the area divided by 1000 (refer to Table 5.6). As was mentioned earlier, the limit state function is explicitly available with respect to the resistances.

Once the closed-form expression for the member internal forces is established, it is possible to do a Monte Carlo analysis of the strength limit state function of each member. Random numbers between zero and one can be generated, and using the inverse cumulative limit state functions of the resistance and the load effect numerous realisations will be extracted and the number of failures will be counted. Dividing the number of failures for each of the performance functions by the total number of simulations will yield the probability of failure. Two types of Monte Carlo analysis were performed: direct Monte Carlo analysis and Latin Hypercube sampling. For the direct Monte Carlo analysis 10,000; 100,000; 1,000,000 and 3,000,000 simulations were performed and the simulation results for each member are presented in Tables 5.16 and 5.17 as shown below.

Table 5.16: Reliability indices and probabilities of failure RSM-DMCS (Case I)

n	10000		100000		1000000		3000000	
members	P_f	β	P_f	β	P_f	β	P_f	β
1	0.0056	2.5364	0.00739	2.4377	0.00657	2.4801	0.00670	2.4730
2	0.0060	2.5121	0.00787	2.4149	0.00704	2.4555	0.00716	2.4494
3	0.0001	3.7190	0.00014	3.6331	0.00014	3.6387	1.62E-04	3.5948
4	0	∞	0	∞	4.00E-06	4.4652	2.67E-06	4.5512
5	0.0056	2.5364	0.00739	2.4377	0.00657	2.4800	0.00670	2.4729
6	0.0032	2.7266	0.00490	2.5828	0.00443	2.6173	0.00449	2.6130
7	0.0073	2.4422	0.00931	2.3531	0.00859	2.3828	0.00873	2.3770
8	0.0001	3.7190	0.00010	3.7190	0.00012	3.6664	1.44E-04	3.6265
9	0.0001	3.7190	0.00023	3.5030	2.37E-04	3.4950	2.68E-04	3.4621
10	0.0072	2.4471	0.00916	2.3591	0.00839	2.3913	0.00854	2.3852

In Table 5.16, the result of the Direct Monte Carlo simulation are given when the load effect follows an extreme value distribution and the resistance a lognormal distribution. In this case since both of random variables are non-normal, the number of failures is counted to calculate the failure probability (Equations 3.35 and 3.36). Then by using Equation 2.12 the reliability index is calculated.

In Table 5.17, the calculation of the results of the Monte Carlo simulation are given where both of the variables are normally distributed. In this case since both of the random variables are normal, the limit state function (G) is also normally distributed. Therefore, it is easier to calculate the mean and standard deviation by using Equations 3.38 and 3.39, and compute the reliability index using Equation 2.17. This way, the results have a reasonable accuracy even with a smaller number of simulations as is evident in Table 5.17.

Table 5.17: Reliability indices and probabilities of failure RSM-DMCS (Case II)

n	1000				10000				100000			
members	μ	σ	β	P_f	μ	σ	β	P_f	μ	σ	β	P_f
1	282.04	82.19	3.43	0.00030	278.51	85.79	3.25	0.00058	278.19	85.56	3.25	0.00057
2	108.89	32.17	3.38	0.00036	107.51	33.58	3.20	0.00068	107.38	33.49	3.21	0.00067
3	54.11	9.40	5.75	4.36E-09	53.68	9.78	5.49	2.03E-08	53.65	9.73	5.51	1.76E-08
4	59.80	7.89	7.58	1.79E-14	59.44	8.18	7.27	1.85E-13	59.41	8.12	7.31	1.30E-13
5	281.28	81.97	3.43	0.00030	277.76	85.56	3.25	0.00058	277.44	85.33	3.25	0.00058
6	270.21	72.92	3.71	0.00011	267.06	76.09	3.51	0.00022	266.77	75.87	3.52	0.00022
7	234.08	72.16	3.24	0.00059	231.00	75.34	3.07	0.00108	230.71	75.15	3.07	0.00107
8	95.43	16.40	5.82	2.93E-09	94.69	17.05	5.55	1.40E-08	94.63	16.96	5.58	1.21E-08
9	63.58	11.60	5.48	2.11E-08	63.06	12.07	5.22	8.75E-08	63.01	12.01	5.25	7.78E-08
10	174.11	53.40	3.26	0.00056	171.83	55.76	3.08	0.00103	171.62	55.62	3.09	0.00101

In Table 5.17 for different number of simulations the mean values and standard deviations are shown for each member. Mean and standard deviation values are rounded up to two digits after the floating point to fit the table. The convergence of the reliability index values and probabilities of failure can be observed in Table 5.17 where the number of required simulations is smaller compared to the case of non-normal random variables (Table 5.16).

The obtained limit state equations are also evaluated using Latin Hypercube Sampling Monte Carlo simulation (LHCSMC). The results are presented in Tables 5.18 and 5.19. For LHCSMC also 1,000; 10,000; 100,000; and 1,000,000 simulations are performed. Again, for non-normal random variables, the failure probability was calculated by counting the number of failures.

Table 5.18: Reliability indices and probabilities of failure RSM-LHCSMC (Case I)

n	1000		10000		100000		1000000	
members	P_f	β	P_f	β	P_f	β	P_f	β
1	0.006	2.5121	0.0065	2.484	0.00654	2.482	0.006556	2.481
2	0.007	2.4573	0.0067	2.473	0.00706	2.454	0.007024	2.456
3	0	∞	0.0002	3.540	0.00014	3.633	0.000151	3.614
4	0	∞	0	∞	0	∞	2.00E-06	4.611
5	0.006	2.5121	0.0065	2.484	0.00654	2.482	0.006559	2.481
6	0.005	2.5758	0.0044	2.620	0.00442	2.618	0.004407	2.619
7	0.009	2.3656	0.008	2.409	0.00863	2.381	0.008637	2.381
8	0	∞	0.0002	3.540	0.0001	3.719	0.000131	3.650
9	0	∞	0.0003	3.432	0.00024	3.492	0.000256	3.474
10	0.009	2.3656	0.008	2.409	0.00847	2.388	0.00843	2.390

Table 5.19: Reliability indices and probabilities of failure RSM-LHCSMC(Case II)

n	1000				10000				100000			
members	μ	σ	β	P_f	μ	σ	β	P_f	μ	σ	β	P_f
1	278.26	86.74	3.21	0.000669	278.19	85.19	3.27	0.0005467	278.18	85.35	3.26	0.000559
2	107.41	33.95	3.16	0.00078	107.38	33.35	3.22	0.0006416	107.38	33.41	3.21	0.000655
3	53.65	9.90	5.42	2.99E-08	53.65	9.69	5.54	1.526E-08	53.65	9.71	5.53	1.64E-08
4	59.41	8.28	7.18	3.57E-13	59.41	8.09	7.35	1.018E-13	59.41	8.11	7.33	1.17E-13
5	277.51	86.51	3.21	0.000669	277.44	84.97	3.27	0.0005469	277.43	85.12	3.26	0.000559
6	266.83	76.95	3.47	0.000263	266.77	75.53	3.53	0.0002064	266.76	75.68	3.52	0.000212
7	230.78	76.17	3.03	0.001223	230.71	74.84	3.08	0.0010248	230.71	74.97	3.08	0.001044
8	94.64	17.26	5.48	2.09E-08	94.63	16.89	5.60	1.047E-08	94.63	16.93	5.59	1.13E-08
9	63.02	12.22	5.16	1.25E-07	63.01	11.96	5.27	6.824E-08	63.01	11.98	5.26	7.29E-08
10	171.67	56.37	3.05	0.001161	171.62	55.38	3.10	0.0009709	171.62	55.48	3.09	0.00099

In Table 5.19 the results for the case of normal random variables are shown. The mean and standard deviation of the limit state function of each member is calculated in order to get the reliability index and probability of failure. This is performed precisely the same way as for the direct Monte Carlo analysis.

Another Method to check the results for the case where both of the random variables are normal is by using Equation 3.2 (First Order Second Moment Method). Thanks to the response surface method, a closed-form linear expression for the limit state functions is available, and since case II is being considered, it can give exact results for the reliability index as far as the obtained limit state functions are concerned. Equation 3.2 is presented here again.

$$\beta = \frac{a_0 + \sum_{i=1}^{n} a_i \mu_i}{\sqrt{\sum_{i=1}^{n} (a_i \sigma_i)^2}}$$

Where a_0 is the intercept of the limit state function, a_i is the factor multiplied by each random variable in the linear limit state equation, μ_i is the mean value of each input random variable (X_i) , and σ_i is the standard deviation of each basic random variable (X_i). Using Equation

3.2 each one of the member limit state functions are evaluated to get the reliability index. The results are presented in Table 5.20 below.

Table 5.20: Reliability indices and probabilities of failure RSM-FOSM(Case II)

Component	β	P_f
1	3.2620	0.00055
2	3.2164	0.00065
3	5.5312	1.59E-08
4	7.3359	1.10E-13
5	3.2618	0.00055
6	3.5280	0.00021
7	3.0799	0.00103
8	5.5967	1.09E-08
9	5.2631	7.08E-08
10	3.0959	0.00098

5.5.5.4 Monte Carlo simulation for the whole structure

Another method that can be used for the component level reliability analysis of the structure is to use simulation methods for the whole structure. The three methods that are used here are direct Monte Carlo, Latin Hypercube sampling Monte Carlo simulation, and Update Latin Hypercube Sampling Monte Carlo. Using these three sampling methods can also provide a good assessment on the efficiency of these methods.

In general, performing a direct Monte Carlo simulation may require a huge number of simulations, and in each of the iterations a deterministic finite element analysis of the whole structure needs to be executed. As a result, it is very time-consuming, if not impossible, to perform the Monte Carlo simulation for the whole structure using the application programming interface of the commercial finite element software Strand7 version 2.4.4. This is simply because the linking between the programming language and the commercial software is rather slow. In order to overcome this, a finite element code is developed and saved as a function in Matlab so that the process becomes less time-consuming. It is also highly recommended to allocate memory by predefining matrices with zeros as their components (using the “zeros” function of Matlab) in case of big simulations. These zeros can then be replaced with numbers generated in the simulation. This is because of the fact that the change in the matrix sizes in each simulation makes program extremely slow as the matrix sizes gets bigger and bigger. The finite element codes for the Monte Carlo simulations as well as the finite element code are available in Appendix II of the thesis.

5.5.5.4.1 Direct Monte Carlo simulation: The direct Monte Carlo analysis was performed by simply extracting realisations form the probability distributions of the random variables and either counting the number of failures (non-normal random variables) or calculating the mean and standard deviations (both random variables normal) following the steps mentioned in Chapter 3. The results are shown in Tables 5.21 and 5.22 below. In Table 5.21 the results are given when the yield stress follows a lognormal distribution and load effect follows a Gumbel distribution. Table 5.22 shows the results when both of the variables are normally distributed.

Table 5.21: Reliability indices and probabilities of failure DMCS (Case II)

n	1000		10000		100000		1000000	
members	P_f	β	P_f	β	P_f	β	P_f	β
1	0.00600	2.512	0.00580	2.524	0.00705	2.455	0.00654	2.482
2	0.00600	2.512	0.00600	2.512	0.00768	2.424	0.00697	2.459
3	0	∞	0	∞	0.00016	3.599	0.00013	3.648
4	0	∞	0	∞	2.00E-05	4.107	7.00E-06	4.344
5	0.00600	2.512	0.00580	2.524	0.00697	2.459	0.00646	2.486
6	0.00500	2.576	0.00370	2.678	0.00481	2.589	0.00428	2.629
7	0.00800	2.409	0.00790	2.414	0.00928	2.354	0.00863	2.381
8	0	∞	0	∞	0.00014	3.633	0.00011	3.686
9	0	∞	0	∞	0.00032	3.414	0.00022	3.512
10	0.00800	2.409	0.00750	2.432	0.00902	2.365	0.00834	2.394

Table 5.22: Reliability indices and probabilities of failure DMCS(Case II)

n	1000				10000				100000			
members	μ	σ	β	P_f	μ	σ	β	P_f	μ	σ	β	P_f
1	280.598	85.221	3.293	0.00050	279.157	85.884	3.250	0.00058	278.078	84.915	3.275	0.00053
2	108.344	33.327	3.251	0.00058	107.779	33.592	3.208	0.00067	107.357	33.212	3.232	0.00061
3	53.840	9.767	5.512	1.77E-08	53.713	9.752	5.508	1.82E-08	53.585	9.657	5.549	1.44E-08
4	59.553	8.187	7.274	1.75E-13	59.475	8.121	7.323	1.21E-13	59.367	8.055	7.370	8.52E-14
5	280.199	84.903	3.300	0.00048	278.765	85.560	3.258	0.00056	277.689	84.595	3.283	0.00051
6	269.292	75.506	3.566	0.00018	268.049	76.003	3.527	0.00021	267.087	75.158	3.554	0.00019
7	232.602	74.847	3.108	0.00094	231.314	75.490	3.064	0.00109	230.370	74.630	3.087	0.00101
8	94.750	16.940	5.593	1.11E-08	94.531	16.908	5.591	1.13E-08	94.309	16.745	5.632	8.89E-09
9	63.317	12.047	5.256	7.38E-08	63.154	12.041	5.245	7.82E-08	62.996	11.921	5.284	6.31E-08
10	173.277	55.317	3.132	0.00086	172.328	55.786	3.089	0.00100	171.630	55.151	3.112	0.00093

5.5.5.4.2 Latin Hypercube Sampling Monte Carlo: One of the sampling methods mentioned in the Chapter 3 was Latin Hypercube Sampling. This method was also used to evaluate the reliability of member performance functions obtained through the response surface method. Here, the method is utilised for the whole structure as follows:

According to the procedures explained in Chapter 3, for each of the random variables a vector containing the realisation of that random variable is formed. The number of realisations (n) is equal to the number of considered intervals. Therefore, if there are n intervals, each realisation vector contains n elements. So here there are two vectors containing n elements: one for the applied load and the other for the yield stress. In order to randomly make n combinations of the elements of these vectors of realisation in a way that each element is used once and only once, vectors containing random permutations of numbers between 1 and n are created. In the case of the ten-bar truss structure two vectors of random permutations are formed for each of the random variables (external force and yield stress). Having all of these vectors of random permutations available, elements of the vectors of realisation are called using their corresponding random permutation vector in order to form the combinations. For instance, if there are 5 intervals, vectors $A = [1, 2, 3, 5, 4]$ and $B = [3, 5, 4, 1, 2]$ are two vectors of random permutations between 1 and 5 for applied load and yield stress respectively. Accordingly, the first combination will consist of the first element from the vector of load realisations and the third element from the vector of yield stress realisations, and so forth. Eventually, for each combination the performance function of each member is analysed implicitly by performing a deterministic finite element analysis, and the number of failures of the member performance functions are counted to get the probability of failure for each member. As noted in the former sections, the probability of failure can be roughly estimated (for the case of non-normal random variables) or precisely approximated (when all the random variables are normal) by estimating the mean and standard deviation of the response.

When implementing the Latin Hypercube sampling, if the number of simulations (or intervals) is large, the same problem as direct Monte Carlo simulation can emerge, and the process can become time-consuming. Hence, the developed Matlab finite element code for the structure will be used when a large number of simulations have to be dealt with.

The results of the Latin Hypercube sampling are shown in Tables 5.23 and 5.24 presented below:

Table 5.23: Reliability indices and probabilities of failure LHCSMC (Case I)

n	1000		10000		100000		1000000	
members	P_f	β	P_f	β	P_f	β	P_f	β
1	0.007	2.457	0.00640	2.489	0.00653	2.482	0.00676	2.470
2	0.007	2.457	0.00680	2.468	0.00694	2.460	0.00723	2.446
3	0.001	3.090	0.00010	3.719	0.00017	3.583	0.00016	3.605
4	0	∞	0	∞	0	∞	1.00E-06	4.753
5	0.007	2.457	0.00630	2.495	0.00645	2.487	0.00669	2.474
6	0.004	2.652	0.00450	2.612	0.00447	2.614	0.00447	2.614
7	0.01	2.326	0.00870	2.378	0.00881	2.374	0.00888	2.371
8	0	∞	0.00010	3.719	0.00012	3.673	0.00014	3.637
9	0.001	3.090	0.00030	3.432	0.00025	3.481	0.00024	3.492
10	0.009	2.366	0.00820	2.400	0.00839	2.391	0.00858	2.383

Table 5.24: Reliability indices and probabilities of failure LHCSM (Case II)

n	1000				10000				100000			
members	μ	σ	β	P_f	μ	σ	β	P_f	μ	σ	β	P_f
1	278.104	86.679	3.208	0.00067	278.063	84.895	3.275	0.00053	278.063	84.923	3.274	0.00053
2	107.367	33.898	3.167	0.00077	107.351	33.204	3.233	0.00061	107.351	33.215	3.232	0.00062
3	53.588	9.906	5.410	3.16E-08	53.585	9.664	5.545	1.472E-08	53.585	9.668	5.543	1.49E-08
4	59.369	8.281	7.169	3.78E-13	59.368	8.067	7.360	9.229E-14	59.368	8.070	7.357	9.42E-14
5	277.715	86.354	3.216	0.00065	277.675	84.575	3.283	0.00051	277.674	84.604	3.282	0.00052
6	267.111	76.773	3.479	0.00025	267.076	75.149	3.554	0.00019	267.076	75.175	3.553	0.00019
7	230.393	76.144	3.026	0.00124	230.356	74.607	3.088	0.00101	230.356	74.631	3.087	0.00101
8	94.314	17.178	5.490	2.01E-08	94.309	16.757	5.628	9.118E-09	94.310	16.764	5.626	9.23E-09
9	63.000	12.222	5.154	1.27E-07	62.996	11.929	5.281	6.42E-08	62.996	11.933	5.279	6.49E-08
10	171.647	56.273	3.050	0.00114	171.619	55.134	3.113	0.00093	171.619	55.153	3.112	0.00093

5.5.5.4.3 Updated Latin Hypercube Sampling method: Updated Latin Hypercube Sampling Method (ULHCSM) is an approach that improves the efficiency of Latin Hypercube

Sampling [16]. The procedure for performing a Latin Hypercube Sampling Monte Carlo (LHC-SMC) was illustrated in Section 5.5.5.4.2. The procedure involves making k vectors of random permutation of integer ranking numbers $P = 1, 2, ..., n$ where n is the number of intervals and k is the number of input random variables. Therefore, matrix P can be a $n \times k$ matrix where each column k is the random permutation of ranking numbers corresponding to k^{th} random variable. The existence of statistical correlation between the columns of matrix P can reduce the efficiency of Latin Hypercube sampling method. As a result, the updated systematic sampling method described in Chapter 3 can be used to reduce the statistical correlation between the columns of P and accordingly, increase the efficiency of Latin Hypercube sampling. The steps of the procedure can be stated as below:

1. Create the random permutation matrix for the input random variables.
2. Use Equation 3.31 to calculate the elements of the Spearman Matrix T corresponding to matrix P.
3. Calculate the lower triangular matrix of the Cholesky fractioning of matrix T which is denoted with Q.
4. Compute the inverse of matrix Q and store it matrix S ($S = Q^{-1}$).
5. Calculate the transformed matrix P which is denoted with P_s.
6. Reorder the elements of each column of matrix P according to matrix P_s.

It should be mentioned that in the updated systematic sampling the first column of matrix P_s will always keep the same ordering as the first column of matrix P. Thus, in each iteration of modification the ordering in other columns will change.

The procedure mentioned above can be seen as a "sub-step" of step 4 of the Latin Hypercube sampling procedure explained in Chapter 3 where n combinations are made from the representative values of the intervals.

The results obtained from the updated Latin Hypercube sampling are shown in Tables 5.25 and 5.26 below. For the case where F follows Gumbel distribution and f_y follows lognormal distribution 200; 1000; 10,000 and 100,000 simulations were performed. Probability of failure is computed by counting the number of failures. For the case where both of the random variables follow a normal distribution the mean and standard deviation are calculated, and utilized to obtain the failure probability and reliability index.

Table 5.25: Reliability Indices and Probabilities of Failure ULHCSMC (Case I)

n	200		10000		100000		1000000	
members	P_f	β	P_f	β	P_f	β	P_f	β
1	0.005	2.576	0.007	2.457	0.0059	2.518	0.00679	2.468
2	0.005	2.576	0.007	2.457	0.0062	2.501	0.00718	2.448
3	0	∞	0	∞	0.0001	3.719	0.00016	3.599
4	0	∞	0	∞	0	∞	1.00E-05	4.265
5	0.005	2.576	0.007	2.457	0.0059	2.518	0.00663	2.477
6	0.005	2.576	0.005	2.576	0.0041	2.644	0.00431	2.627
7	0.005	2.576	0.009	2.366	0.0076	2.428	0.00875	2.376
8	0	∞	0	∞	0.0001	3.719	0.00015	3.615
9	0	∞	0	∞	0.0003	3.432	0.00024	3.492
10	0.005	2.576	0.009	2.366	0.0073	2.442	0.00845	2.389

Table 5.26: Reliability Indices and Probabilities of Failure ULHCSMC (Case II)

n	1000				10000				100000			
members	μ	σ	β	P_f	μ	σ	β	P_f	μ	σ	β	P_f
1	277.734	86.061	3.227	0.00063	278.141	85.252	3.263	0.00055	278.061	84.871	3.276	0.00053
2	107.224	33.655	3.186	0.00072	107.382	33.343	3.221	0.00064	107.350	33.194	3.234	0.00061
3	53.527	9.869	5.424	2.91E-08	53.590	9.711	5.518	1.71E-08	53.585	9.661	5.547	1.46E-08
4	59.306	8.282	7.161	4.02E-13	59.370	8.108	7.322	1.22E-13	59.368	8.064	7.362	9.04E-14
5	277.346	85.740	3.235	0.00060	277.752	84.931	3.270	0.00054	277.672	84.552	3.284	0.00051
6	266.765	76.251	3.499	0.00023	267.142	75.472	3.540	0.00020	267.074	75.127	3.555	0.00019
7	230.080	75.586	3.044	0.00117	230.426	74.915	3.076	0.00105	230.354	74.586	3.088	0.00101
8	94.208	17.116	5.504	1.86E-08	94.318	16.839	5.601	1.07E-08	94.309	16.751	5.630	9.01E-09
9	62.928	12.171	5.170	1.17E-07	63.003	11.986	5.256	7.34E-08	62.996	11.924	5.283	6.36E-08
10	171.414	55.862	3.069	0.00108	171.671	55.363	3.101	0.00096	171.618	55.119	3.114	0.00092

5.5.6 Comparison and investigation on the result of reliability evaluation

The reliability evaluation of the ten-bar truss structure was performed using three different methods, and two cases were considered: one is when the load effect follows an Extreme Type I distribution while the yield stress is lognormally distributed (Case I); in the second case all of the random variables are normally distributed (Case II). Firstly, the reliability evaluation was performed using the developed algorithm of the FORM reliability analysis based on the Newton-Raphson recursive process using finite difference sensitivity approach. The finite element analysis results were fed into the reliability analysis algorithm using the application programming interface (API) of a commercial finite element analysis software package, Strand7 version 2.4.2. Also, a fully stochastic finite element formulation was used to perform the reliability analysis where alterations to the finite element code were applied for the purpose of reliability analysis. Secondly, the response surface method was used to get a closed-form (explicit) expression for the limit state function of each member of the structure. Two types of simulation methods were utilized to evaluate the obtained closed-form limit state equations and get the probability of failure and reliability index of each member of the truss: direct Monte Carlo simulation (DMCS) and Latin Hypercube sampling Monte Carlo (LHCSMC). Finally, different methods of simulation for the whole structure were used to get the probability of failure for each member of the truss. These methods included direct Monte Carlo simulation, Latin Hypercube sampling

Monte Carlo, and updated Latin Hypercube sampling Monte Carlo.

5.5.6.1 Comparison between the results of different methods

The comparison between different methods of reliability evaluation is presented in Table 5.27 below. For all the cases of simulation the number of simulations that is used to compare the methods is 100,000.

Table 5.27: Comparison of reliability index results for Case I

Method	FORM		Response Surface		Monte Carlo Simulation		
members	*FD−FSEM*	*FSFEM*	*DMCS*	*LHSSMC*	*DMCS*	*LHCSM*	*ULHCSM*
1	2.485	2.451	2.438	2.482	2.455	2.482	2.468
2	2.459	2.429	2.415	2.454	2.424	2.460	2.448
3	3.627	3.485	3.633	3.633	3.599	3.583	3.599
4	4.538	4.190	∞	∞	4.107	∞	4.265
5	2.485	2.455	2.438	2.482	2.459	2.487	2.477
6	2.626	2.590	2.583	2.618	2.589	2.614	2.627
7	2.386	2.354	2.353	2.381	2.354	2.374	2.376
8	3.650	3.519	3.719	3.719	3.633	3.673	3.615
9	3.499	3.374	3.503	3.492	3.414	3.481	3.492
10	2.394	2.367	2.359	2.388	2.365	2.391	2.389

In Table 5.28 the comparison between the obtained reliability indices are given for Case II where all the basic random variables are considered to be normally distributed. Here the number of simulations for comparison is 10,000.

Table 5.28: Comparison of reliability index results for Case II

Method	FORM		Response Surface		Monte Carlo Simulation		
members	*FD−SFEM*	*FSFEM*	*DMCS*	*LHSSMC*	*DMCS*	*LHCSM*	*ULHCSM*
1	3.263	3.260	3.246	3.265	3.250	3.275	3.276
2	3.216	3.218	3.201	3.220	3.208	3.233	3.234
3	5.529	5.518	5.489	5.538	5.508	5.545	5.547
4	7.337	7.326	7.266	7.346	7.323	7.360	7.362
5	3.264	3.268	3.246	3.265	3.258	3.283	3.284
6	3.530	3.537	3.510	3.532	3.527	3.554	3.555
7	3.080	3.073	3.066	3.083	3.064	3.088	3.088
8	5.576	5.600	5.553	5.604	5.591	5.628	5.630
9	5.267	5.255	5.224	5.270	5.245	5.281	5.283
10	3.096	3.098	3.082	3.099	3.089	3.113	3.114

5.5.6.2 Investigation on the first order reliability analysis methods

The convergence process of the two proposed SFEM algorithms are shown to demonstrate the efficiency and robustness of the algorithm as per Table 5.29 to 5.32. For the FD-SFEM method the total number of iterations was 51 where it took 18 minutes and 27 seconds for the algorithm to complete for Case I. For Case II total number of iterations was 20 and it took 8 minutes and 49 seconds to complete the algorithm. On other hand, for the fully stochastic finite element calculation, for case I there were 50 iterations and for case II there were 30 iterations. The whole process only took seconds for both of the cases (less than 5 seconds). The small difference between the two methods in the number of iterations is due to the slightly different programming where for FSFEM the iterative procedure was performed for the whole structure at once whereas for the FD-SFEM method the iterative procedure was performed separately for each member (refer to Appendix II). It should be noted that all of the calculations were done on a Intel Core i5 machine. The result at the end of each iterative process can be compared with the ones presented in Tables 5.27 and 5.28.

Tables 5.29 to 5.32 show the convergence process for the members of the truss structure for Case I and II. From the tables it can be seen that the convergence is in general quite fast. In fact, for almost all the components in both cases the iteration tends to converge from the second iteration on. The only reason that the FORM analysis continues for more iterations is the strict convergence criterion of 0.0001. Clearly, selection of a less strict criterion can lead to reducing the computation time with an acceptable accuracy. It also shows that selecting the mean value of the basic input random variables can be an appropriate starting point for the iteration. It is hence recommended that such point be defined as the default starting point for a computerised reliability evaluation environment based on SFEM. It is also observed that for the case of normal random variables the convergence is fast where the maximum number of iteration doesn't exceed 2 or 3 iterations for each component. This can be because of the fact that there is no need to calculate the equivalent normal parameters based on the design points obtained in each iteration. In short, the data presented in tables verify the fact that the FORM method can be a very fact and efficient method for the component level reliability analysis of the truss structures with a good level of accuracy regardless of whichever method is used.

Table 5.29: Iteration process for FD-SFEM Case I

	RI in each iteration					
members	1	2	3	4	5	6
1	3.61570	2.56573	2.48634	2.48305	2.48302	N/A
2	3.57186	2.54051	2.46355	2.46043	2.46040	N/A
3	5.93711	3.91298	3.65175	3.62211	3.62161	N/A
4	7.74367	5.08086	4.63282	4.53474	4.53163	4.53159
5	3.62376	2.57036	2.49052	2.48720	2.48717	N/A
6	3.90408	2.73106	2.63466	2.63004	2.62999	N/A
7	3.42081	2.45335	2.38448	2.38190	2.38188	N/A
8	6.02095	3.96381	3.69436	3.66281	3.66226	N/A
9	5.67085	3.75318	3.51744	3.49327	3.49290	N/A
10	3.44692	2.46845	2.39821	2.39555	2.39553	N/A

Table 5.30: Iteration process for FD-SFEM Case II

	RI in each iteration	
members	1	2
1	3.26339	3.26339
2	3.21571	3.21571
3	5.52949	5.52949
4	7.33732	7.33732
5	3.26386	3.26386
6	3.52977	3.52977
7	3.07969	3.07969
8	5.57588	5.57588
9	5.26698	5.26698
10	3.09602	3.09602

Table 5.31: Iteration process for FSFEM Case I

	RI in each iteration				
members	1	2	3	4	5
1	3.10567	2.45706	2.45038	2.45066	2.45064
2	2.88016	2.45568	2.42868	2.42912	2.42910
3	4.88006	3.42810	3.48191	3.48490	3.48453
4	6.78114	3.50932	4.18612	4.19151	4.19024
5	3.00569	2.47199	2.45425	2.45462	2.45460
6	3.41356	2.58683	2.58969	2.59007	2.59004
7	3.13731	2.34528	2.35392	2.35409	2.35408
8	6.07019	3.19045	3.51177	3.51957	3.51885
9	4.60345	3.35247	3.37184	3.37454	3.37426
10	2.75526	2.39579	2.36670	2.36710	2.36709

Table 5.32: Iteration process for FSFEM Case II

	RI in each iteration		
members	1	2	3
1	2.85198	3.25976	3.25976
2	2.63413	3.21774	3.21774
3	4.60873	5.51775	5.51775
4	6.50851	7.32552	7.32552
5	2.75577	3.26751	3.26751
6	3.15283	3.53683	3.53683
7	2.87983	3.07311	3.07311
8	5.76467	5.60048	5.60048
9	4.33472	5.25527	5.25527
10	2.51222	3.09801	3.09801

5.5.6.3 Investigation on simulation methods

Another important issue is investigating the efficiency of the proposed simulation methods. In order to investigate the efficiency, the reliability index for member 1 is calculated 10 times for each of the simulation methods. Each time 100,000 simulations will be performed and the reliability index results for member 1 is recorded. Finally, the mean value and standard deviation are computed for each of the methods. It should also be noted that the updated Latin Hypercube sampling method is implemented with two updating iterations (u=2). The results are presented in Table 5.33 for Case I where load effect variable is extreme type I (Gumbel) and resistance variable is lognormal.

Table 5.33: Comparing the efficiency of different suggested simulation methods

	Reliability index values		
Computation No.	*DMCS*	*LHCSMC*	*ULHCSMC*
1	2.4949	2.4522	2.437236
2	2.3867	2.4573	2.478327
3	2.4677	2.5491	2.452164
4	2.4422	2.4893	2.462428
5	2.5364	2.4624	2.447127
6	2.5622	2.4522	2.472958
7	2.4893	2.4324	2.478327
8	2.4044	2.4949	2.462428
9	2.4838	2.5063	2.467658
10	2.4893	2.4624	2.432379
Mean	2.4757	2.4758	2.4591
S.D	0.00291	0.00117	0.000268

The effect of increasing the number of simulations on the mean value and the standard deviation of the calculated reliability index using suggested simulation methods can also be investigated. This will help evaluate the efficiency of the proposed simulation methods as the number of simulations increases. This investigation is performed for the updated Latin Hypercube sampling. Different numbers of simulations are considered and each time the mean and the standard deviation of the computed reliability index for member 1 of the truss structure is obtained. The results are recorded and graphs are produced based on the data.

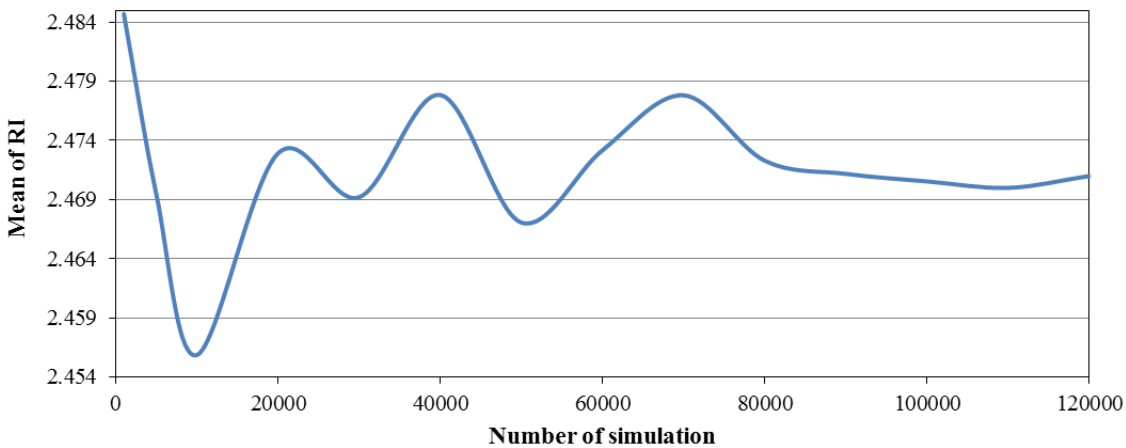

Figure 5.7: Mean of reliability index vs number of simulations

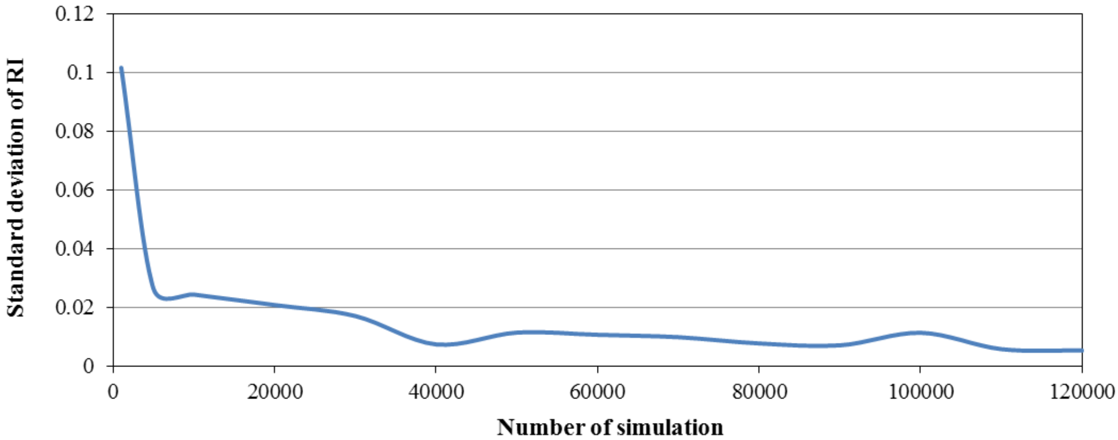

Figure 5.8: Standard deviation of the reliability index vs number of simulations

In Figure 5.7 the mean of reliability index versus the number of simulations is shown. The number of simulations that were considered are in the interval of [1000, 120000]. It is clear from Figure 5.7 that as the number of simulations passes 80,000 the curve tends to become flatter.

Figure 5.8 shows the effect of increasing the number of simulations on the standard deviation of the obtained reliability index. It can be seen that the increase in simulation numbers considerably decreases the standard deviation of the computed reliability index.

Another important issue regarding the simulation methods is the failure of these methods in calculation of the reliability index or the failure probability for members with high reliability indices or low probabilities of failure. For the case of normal random variables the second method was used for reliability calculation where the mean and standard deviation were obtained to get the reliability index. However, if the number of failures had to be counted, a large number of simulations would be required. This can be investigated using Equation 3.40 presented in Chapter 3. By manipulating Equation 3.40, an estimate for the required number of simulations corresponding to a certain level of accuracy can be obtained:

$$N = \frac{1 - P_f^T}{e^2 \times P_f^T} \times 200^2 \tag{5.12}$$

In Equation 5.12, e is the percentage of error and P_f^T is the true probability of failure. If the failure probabilities calculated through the developed FSFEM algorithm are considered to be the true failure probabilities, then it is possible to get the required number of simulations for different levels of accuracy.

Table 5.34 shows the number of simulation needed for 20%, 15%, 10%, and 5% error for the direct Monte Carlo Simulation. This is presented for members 3, 4, 8, and 9 which have high reliability indices or low failure probabilities.

Table 5.34: Number of simulations needed for members with low failure probabilities

		Percentage Error			
Members	P_f^T	20%	15%	10%	5%
3	1.72E-08	5.82E+09	1.04E+10	2.33E+10	9.32E+10
4	1.19E-13	8.40E+14	1.49E+15	3.36E+15	1.34E+16
8	1.07E-08	9.36E+09	1.66E+10	3.74E+10	1.50E+11
9	7.39E-08	1.35E+09	2.41E+09	5.41E+09	2.16E+10

As is clear from Table 5.34, for member 3, 4, 8, and 9 a large number of simulations is needed. It should be mentioned that the largest number of simulations that was performed was in the order of 10^6 whereas according to Table 5.30 even with an accuracy level of 20% error the number of simulations that are needed for member 4 is in the order of 10^{13}, and for the other components (3, 8, and 9) in the order of 10^8. This very well proves the inefficiency of the simulation methods in computation of reliability (or failure probability) for components with high reliability indices or low probabilities of failure.

5.5.6.4 Investigation on the effect of χ on the reliability of compression members

In Section 5.5.3 it was mentioned that in the reliability evaluation of compression members the coefficient χ was treated as a deterministic value for the simplification of calculations, and the fact that its effect on the reliability of members was not significant. This is demonstrated by a reliability calculation where χ is considered in terms of yield stress which in turn is considered as a random variable. Two type of component reliability calculations are used to investigate the effect. First, a fully stochastic finite element method is used. In this case all the calculation are the same as 5.5.5.2, except the derivative of R in terms of yield stress is different in this case.

$$R(f_y) = A \times f_y \times (1 + \lambda^{2n})^{-1/n}$$

where

$$\lambda = \frac{KL}{r\pi}\sqrt{\frac{f_y}{E}}$$

To calculate the derivative of resistance with respect to the yield stress, it can be assumed that $R = X \times Y$ where $X = A \times f_y$ and $Y = (1 + \lambda^{2n})^{-1/n}$. The derivative of R can be shown as below.

$$\frac{\partial R}{\partial f_y} = \frac{\partial X}{\partial f_y} \times Y + \frac{\partial Y}{\partial f_y} \times X \tag{5.13}$$

After computation Equation 5.13 yields,

$$\frac{\partial X}{\partial f_y} \times Y = A \times (1 + \lambda^{2n})^{-1/n} \tag{5.14}$$

$$\frac{\partial Y}{\partial f_y} \times X = -(\frac{KL}{r\pi})^2 \times \frac{1}{E} \times (\lambda^2)^{n-1} \times (1 + \lambda^{2n})^{\frac{-1-n}{n}} \times A \times f_y \tag{5.15}$$

As a result, the derivative of the resistance with respect to the yield stress can be formulated as shown below.

$$\frac{\partial R}{\partial f_y} = A \times (1 + \lambda^{2n})^{-1/n} - (\frac{KL}{r\pi})^2 \times \frac{1}{E} \times (\lambda^2)^{n-1} \times (1 + \lambda^{2n})^{\frac{-1-n}{n}} \times A \times f_y \tag{5.16}$$

Evidently, Equation 5.16 is used at the design point to calculate the derivative of resistance with respect to the yield stress (f_y) at each iteration of the FORM method.

The second method that is used is a direct Monte Carlo simulation method to verify the results that are obtained through FSFEM. The difference here is that in each simulation of DMCS a new value for χ based on the new generated value of yield stress is obtained whereas previously the value of χ was kept constant throughout the simulation. The results of these two methods are presented and compared for both of the cases. Tables 5.35 and 5.36 show the comparison for Cases I and II using a fully stochastic finite element method as well as direct Monte Carlo simulation. Different reliability index and probability of failure results are shown for the compression members of the structure.

Table 5.35: Comparison between χ as a deterministic value (DV) and as a random variable (RV)-SFEM

Method	Case I				Case II			
Method	DV		RV		DV		RV	
members	β	P_f	β	P_f	β	P_f	β	P_f
1	2.4506	0.00713	2.3090	0.01047	3.2598	0.00056	3.0675	0.00108
2	2.4291	0.00757	2.2323	0.01280	3.2177	0.00065	2.9290	0.00170
6	2.5900	0.00480	2.3965	0.00828	3.5368	0.00020	3.2667	0.00054
9	3.3743	3.70E-04	3.1581	7.94E-04	5.2553	7.39E-08	5.1083	1.62E-07

Table 5.36: Comparison between χ as a deterministic value (DV) and as a random variable (RV)-DMCS

Method	Case I				Case II			
Method	DV		RV		DV		RV	
members	β	P_f	β	P_f	β	P_f	β	P_f
1	2.4815	0.00654	2.3072	0.01052	3.2748	0.00053	3.0609	0.00110
2	2.4590	0.00697	2.2315	0.01283	3.2325	0.00061	2.9258	0.00172
6	2.6292	0.00428	2.3938	0.00834	3.0868	0.00101	3.0744	0.00105
9	3.5124	0.00022	3.1755	7.48E-04	5.2844	6.31E-08	5.1161	1.56E-07

The results shown above clearly prove that the effect of assuming χ as a deterministic variable can cause an overestimation of reliability index value of compression members. As a result, in order to obtain a more realistic estimation of reliability index value and failure probability, χ needs to be considered as a random variable.

5.5.7 Conclusions on the methodologies

In previous sections, the reliability evaluation of a statically indeterminate ten-bar truss was performed using three methods of reliability calculation. Two different cases were considered for the structure. In Case I, load effect follows an Extreme Type I distribution and the resistance (yield stress) has a lognormal distribution. In case II, both of the random variables are normally distributed. In both of the cases, all of the basic random variables were uncorrelated.

The following conclusions are presented regarding a component level reliability analysis:

- Computer programs were developed for all of the aforementioned methodologies. It was shown that all the methods can be properly used for a computerised component level reliability analysis. For all of the above-mentioned methods, it is essential that proper interaction between the reliability evaluation modules and the deterministic finite element analysis be established. In the case of simulation methods, the results of the deterministic FEM analysis are used for the simulation and computation of failures during the simulation. For response surface method, FEM analysis is needed to obtain the required values for the regression and formation of the explicit limit states, and for the First Order Reliability Methods (FORM) the FEM analysis is exploited for the calculation of the limit state function derivatives either through a finite difference method or through a classical perturbation where alterations to the finite element coding are necessary. As a result, failure to establish a proper link to the finite element module will lead to an incorrect and inaccurate reliability evaluation of the components.

- Among all the simulation methods, the updated Latin Hypercube seems to be the most efficient one. This fact is supported by the data presented in Table 5.33. The only problem with the updated Latin Hypercube sampling is the fact that the updating process and re-arrangement of the permutation matrix can make the process more time-consuming. Thus, if applying the updated Latin Hypercube sampling is not possible, the Latin Hypercube sampling can be a proper replacement. The Latin Hypercube sampling can take less time compared to the updated Latin Hypercube sampling, and it is more efficient compared to the direct Monte Carlo simulation. However, direct Monte Carlo method can also be useful for reliability computation.

- The response surface method was also used for the reliability assessment. With respect to component level reliability evaluation, response surface methods might not be as efficient as the other two methods since it is not as straightforward to program, and it has to be used together with either a simulation method or a first order reliability method anyway. However, when linear elastic analysis is performed and strength limit states of the components are being investigated, the regression part of the method can be eliminated. This is due to the fact that the relationship between the load and displacements is linear and only by applying an external force vector of unit forces, such as F_I (refer to step 5 of FSFEM method in Section 5.5.5.2), the relationship between the external and internal component forces can be established. In other terms, the factor m shown in Section 5.5.5.3 can be obtained this way. This method can specifically be efficient in calculation of the system reliability. This will be discussed in Chapter 7 comprehensively.

- For the case of FORM methods, a fully stochastic finite element formulation seems more efficient compared to the application of a commercial FEM package such as Strand7. The fully stochastic finite element method only takes less than 5 seconds to compute the component reliability whereas the proposed FD-SFEM methodology takes about 1200 seconds for case I and about 500 seconds for Case II. However, both methods prove to yield reasonably accurate results.

- In general, it is recommended that a finite element code be provided instead of using a commercial FEM package. This is due to two reasons: first, the efficiency of using a commercial FEM package is significantly less than a developed FEM code, and, Moreover, due to the lack of speed using a commercial FEM package is not a viable option for simulation methods; therefore, a finite element code has to be provided for simulation methods anyway (a FEM code that is developed in the same programming language as the reliability analysis modules). Second, The coding of an application programming interface (API) that is used as a link between the commercial FEM package and the reliability analysis modules appears no less complex than developing a FEM code. Therefore, a method such as FD-SFEM is only recommended when the finite element formulation is rather complex.

- In reliability evaluation of compression members the value of χ, which further reduces the member resistance, is recommended to be considered as a random variable. This will lead to a more realistic evaluation of the reliability index where it was shown that the effect in some cases can be significant.

- In conclusion, it can be inferred that for a comprehensive component level reliability evaluation all the three methods are useful and access to all the developed methods should be provided in an integrated reliability analysis environment. A more sophisticated method such as FSFEM can be used for the preliminary evaluation which offers a faster way of reliability computation, and a simulation method can then be used for the confirmation of the FSFEM results. The response surface methods, should also be included in an integrated reliability environment where its usefulness is employed in system reliability calculation.

Chapter 6

Background to System Reliability Evaluation of Truss Structures

6.1 Introduction

6.1.1 General

In the previous chapters different methods of calculating the component reliability of truss structures were discussed and in Chapter 5 computerised application of these methods was investigated on a ten-bar truss structure. In this chapter the objective is to investigate the implementation of different methods of system reliability calculation for truss structures.

System reliability has been a topic of research for a long time. Specifically, from the 1980's there has been a lot of research done in this area, e.g. [31,34,51]. In general, system reliability analysis of structures is a rather complex subject and so far it has been mostly developed for idealized structures, and still there is a need for research in this area.

Despite all of the difficulties mentioned above, system reliability is an important factor in reliability evaluation of structures. The overall system reliability of a structure can be significantly different form the component reliabilities [34]. As a result, in order to obtain a genuine evaluation of a structural system the calculation of system reliability can be of paramount importance. The issue of system reliability can also arise when the repair optimization of a structure (for instance, a bridge structure) is intended. Since it may reveal that some repair actions are more important than others. It could also show that if each individual component of a structure is safe the whole system may still be unsafe [13].

Considering all of the aforementioned remarks, it is essential that the system reliability of truss structures be investigated so that a better idea of the safety of the whole structure as a system can be obtained. Generally, computation of system reliability and defining proper failure modes is dependent on the material resistance model and behaviour of the elements. The material can be assumed elasto-plastic and the failure can be brittle, semi-brittle or ductile. All of these factors are crucial in the determination of system reliability. All of these matters will be discussed in the following sections.

6.1.2 Background to system reliability analysis of trusses

In general, there are different proposed methods for system reliability analysis of (truss) structures. These methods suggest various approaches for identification and generation failure sequences that are used for system reliability calculation. These methods include methods that focus on deterministic identification of failure sequences where there is no need for the stochastic analysis of the structure for identification of the failure modes. In fact, these methods use the member utilization ratios for the generation of failure sequences which are obtained through a deterministic analysis. For instance, the incremental loading method can be categorized as a method that uses a deterministic approach for the failure mode identification [23, 34, 40]. On the other hand, some other methods use the stochastic evaluation of the structure for the generation of the failure sequences. In this case, a stochastic analysis of the structure is used and the most likely failure sequences are generated based on the results of the reliability analysis of the structure at different damaged “stages” of the structure. Examples of these methods are the branch and bound method and the β-unzipping method [6, 23, 45]. These methods utilise the result of the reliability analysis for the failure sequence generation; however, they use different criteria to achieve this. With regards to truss structures there are two issues that are arising for the proposed methodologies:

- All of these proposed methodologies look at system reliability from a general point of view and not specifically for truss structures. Therefore, they don’t specifically address the issues that arise in system reliability analysis of truss structures.
- They don’t investigate the algorithmic system reliability analysis of trusses. In other term, proper algorithmic approaches that can be used for computer programming of system reliability analysis of truss structures are not addressed in the proposed methodologies.

Based on the aforementioned points, truss structures need to be looked at specifically and in more detail where the algorithmic approaches to system reliability analysis of these structures is investigated based on the proposed methodologies. In the following sections of the thesis these issues will be addressed and in Chapter 7 detailed algorithmic analysis of truss structures will be investigated. It should also be noted that the emphasis is the on the methodologies that use a stochastic analysis of the structure for failure mode identification. As a result, methods such as the β-unzipping method and the branch and bound method will be utilised.

6.2 Material modelling in system reliability calculation

Material models can play an important role in calculation of the system reliability for a (truss) structure. Generally, the actual material behaviour can be very complex, and consequently simpler models of material behaviour are required. One of the usual ways to make this simplification is to assume a bilinear axial force-deformation model for the material [23]. With this assumption the non-linearity of the material behaviour can be referred to as a two-state material model. Therefore, the material will go through two linear states.

Another important matter regarding the behaviour of the material is the type of failure it will follow after it has reached its capacity. The failure is mostly considered to be either ductile or brittle, and sometimes a semi-brittle failure case can also be considered. For instance, an off-shore jacket structure can be modelled as a truss structure where the diagonal element in compression can be assumed to have semi-brittle failure and diagonal members in tension can be elastic-plastic (ductile). These types of material behaviour are shown in Figure 6.1 to 6.3.

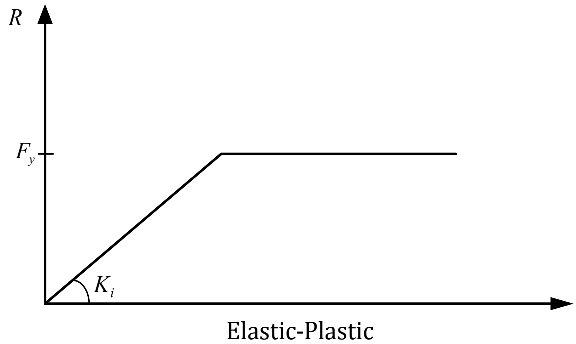

Figure 6.1: Elastic-Plastic behaviour [32]

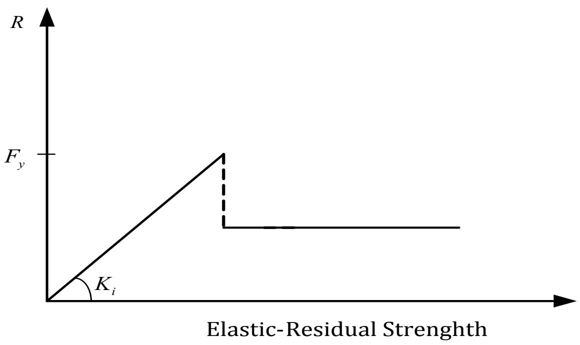

Figure 6.2: Elastic- Residual Strenghth material behaviour (semi-brittle) [32]

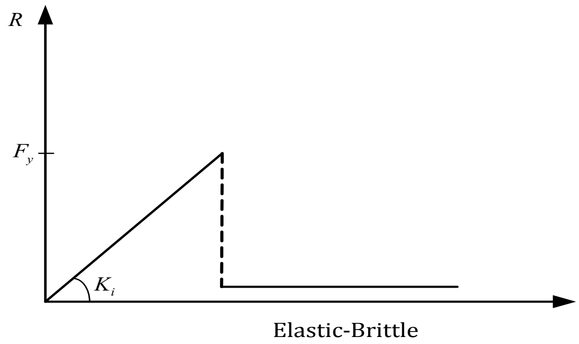

Figure 6.3: Elastic-Brittle behaviour [32]

It is obvious from Figures 6.1 to 6.3 that all the materials that are modelled this way can only be in two states. The first state is the elastic state where the element has the stiffness K_i. This state is also called a *safe state*. The second state is the flat line portion in the load displacement graph and the stiffness of the element in this partition is zero. This state is called *failed state*. Other types of element behaviour can also be considered such as elastic-hardening behaviour or curvilinear behaviour [32]; however, these types of material behaviour are not considered here due to their complexity.

It can also be observed that ductile elements can sustain the load carrying capacity when they enter the failed state. On the contrary, brittle elements will cease to carry load after they yield (or "fail"). Semi brittle elements will continue to carry load as they reach the failed state except that they won't be able to maintain their full load-carrying capacity.

6.3 Other important assumptions regarding structural behaviour

There are some other significant assumptions that are necessary to be mentioned regarding the system reliability calculation of truss structures:

1. All the loads applied on the truss structure are assumed to be static and the response of the structure is obtained by performing a static linear finite element analysis, thus the dynamics are ignored.

2. Second order effects are neglected (such as geometrical non-linearity and P-δ effects).

3. Elements are not supposed to have strain reversal, so if an element is failed (in failure state), it will remain in a failure state.

4. Stiffness of the non-linear elements are assumed to be deterministic in the linear static analysis.

5. Resistances of the elements can be considered as random variables.
6. Loads on the nodes of the structure follow a fixed load pattern. It means all of the loads are proportional.

The assumptions noted above are of paramount importance in system reliability analysis for the proposed methods. If there are other necessary assumptions regarding a specific methodology, they will be mentioned within the procedure of that methodology.

6.4 Methods of system reliability computation

6.4.1 Introduction

There are some methods proposed to approximate the system reliability of structures or at least to define some bounds for the system reliability. These methods are mainly based on a so-called failure path approach [35, 47]. These methods determine the system failure as the failure of a set of its elements, and it leads to a cut-set formulation for the system reliability evaluation [23]. This method will be discussed more in detail in the following sections.

Another method is the proposed β-unzipping method [45]. It also uses the failure path of the system or works with the sequences of element failures and calculation of reliability index at each state of the structure. In this method, the reliability is evaluated at different levels and at each level a certain assumption is made regarding the system failure. The β-unzipping method assumes that all the basic random variables are normally distributed. So if the random variables do not follow a normal distribution, certain methods are proposed for the reliability calculation (such as FORM). This approach will also be discussed in detail in Section 6.4.4.

One other method to estimate the system reliability is the branch and bound method. This method is a method that is based on a branch and bound search to identify the failure paths of the system. The identification of the failure paths is achieved through the evaluation of the structure in different damaged states which correspond to a state that is most likely to happen. Through the analysis, the failure tree of the structure is expanded and damaged states are formed based on the failure tree expansion of the whole system. This method will be discussed in Section 6.4.3.1.

6.4.2 Failure path approach

As mentioned in the former section, failure path approach considers structural failure as a sequence of failure of its elements. Each set of failed elements is called a cut-set event. For a truss structure failure elements are components of the truss where elements can either fail in tension or compression. The failure of the whole truss can be considered as the collapse of the structure under the applied loads. Elements are safe when they are in the elastic region of the elastic-plastic model and are failed if they are in the flat region where the stiffness is zero.

The whole process of failure can be showed with the help of a failure tree. S^0 is considered as the state where all of the elements are in the safe region (elastic region), and the structure is undamaged. A damaged state of the structure can be shown if elements i and j have failed, and it can be denoted as S^{ij}. E_k^{ij} can be taken as the event that element k fails next, given that elements i and j have already failed. Consequently, in order to be able to construct the failure tree each node can be deemed as the current state of the structure and each branch of the failure tree can be seen as an event.

The initial node in the failure tree demonstrates the undamaged state of the structure. At this state, a number of failure events can occur for some of the elements of the structure. Thus, each failure event can be one branch emitting form the initial node. At the end of each of these new branches there is a node. If these new nodes or states are not representing a collapse state for the structure, then from each of these nodes some branches can again be emitted (some elements can fail next). This process is continued until a collapse state of the system is reached. If a node is in a structural collapse state in can be referred to as a *terminal node.*

Having the failure tree completely formed, different failure sequences for the structure can be defined as the paths from the initial node to each of the terminal nodes. Each one of these sequences is a cut-set event as was mentioned earlier. System failure is defined as the occurrence of any of the cut-set events (or failure paths in other terms). Assume the failure of the structure happens in cut-set event l which consists of i events, the system failure probability can be mathematically expressed as in Equation 6.1 [23].

$$P\{\text{any cut set event } l\} = \mathrm{P}\{\cup_l \cap_{l_1,l_2,..l_i} E_{l_i}^{l_1,l_2...,l_{i-1}}\} \tag{6.1}$$

The failure tree for the three-bar structure shown in figure 6.4 can be considered as a good example to better clarify the procedure [23]:

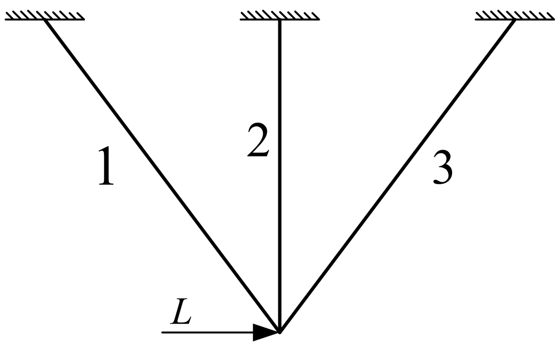

Figure 6.4: A three-bar structure

The failure tree for the structure is shown in Figure 6.5. The nodes (states) and branches (events) can be seen in the figure. It starts from the initial state where the structure is undamaged and none of the elements are in the failure region and ends with terminal nodes which represent the state where the structure has collapsed. Different failure sequences (failure paths) or cut-set events are clearly shown on failure tree. In the failure tree of Figure 6.5, there are 6 failure paths

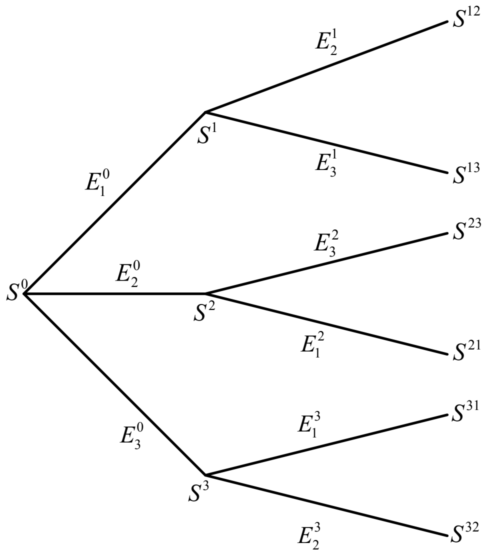

Figure 6.5: Failure tree for the three-bar structure

for the structure. For instance, state S^{23} shows that the cut-set event referring to the failure of elements 2 and 3 can cause the collapse of the structure. It should be noted that the sequences of element failures are important and for example, states S^{23} and S^{32} are not the same since they follow different paths albeit they might refer to the same physical state of the damaged structure.

6.4.3 Defining the failure paths

As it was noted before, in the failure path approach the probability of failure for the whole system is defined as the occurrence of any of the failure sequences (failure modes). For large structures, however, there are a large number of failure sequences and trying to perform a complete enumeration of all those failure sequences could be a very cumbersome task, if not impossible. Furthermore, in most of the cases only a small fraction of failure sequences contribute to the overall probability of failure of the system [6, 7]. Therefore, some methods need to be used in order to determine these important failure sequences. These types of methods will lead to a lower bound to the system probability of failure since they only consider a subset of failure sequences [23].

There are different proposed methods that try to identify these important failure modes or failure sequences for system reliability evaluation of structures. These methods can be divided into two categories: The first category includes methods that function with respect to the mean values of the basic random variables and in a sense they could be deemed as more deterministic than stochastic. The second category, however, includes methods that possess a probabilistic nature.

First category: these methods are mean-based failure sequences, Monte Carlo simulation for failure sequence generation, and utilization-ratio-based failure sequences. All of these methods, as is evident from their names, work with random variables at their mean values. This can lead to the generation of unimportant failure sequences. This inaccuracy is due to the fact that failure usually happens when load variables are at higher values and resistance variables are at their lower values which are mostly away from mean value of the random variables or, in other terms, are at low probability regions. As a result, failure sequences that are produced by these methods might not be the most likely failure sequences to occur [23].

Second category: marginal-failure-probability-based sequences, and branch and bound methods can be placed in this category. These approaches generate the failure sequence based on the probability of failure of different element. Therefore, they can usually lead to the generation of more important failure sequences. Probabilistic failure generation methods usually are computationally more difficult compared to mean-based failure sequence generation methods.

Among all the methods discussed above the focus will be on branch and bound method and it will be applied to a truss structure. Other methods can be compared with this method.

6.4.3.1 Branch and Bound Method

The Branch and bound method is a probabilistic method that is useful in finding the most probable failure sequences for a structure. In this method each step refers to a damaged state of the structure. At each stage the probabilities of failure for failure elements has to be computed carefully.

The procedure starts with the calculation of failure probabilities for each of the failure elements at the undamaged state of the structure. For instance, for a truss structure, initially, the probability of failure for each of the members of the intact truss structure (undamaged structure) has to be found. Let's assume it is possible to have n failure element for the intact structure (for a truss each member is a failure element; therefore, if a truss has n elements, for the intact structure there could be n damaged states). The probability of failure that leads to each of these damage states has to be computed. For instance, if element i is to fail, the probability of failure is shown as $P(E_i^0)$ (It means the probability that element i fails in the intact structure is $P(E_i^0)$). This calculation has to be done for all of the other n-1 failure elements. At this stage the nodes corresponding to each of the n failure states of the structure are formed, and they are called external nodes. Now the node that corresponds to the intact structure is called an internal node (node corresponding to S^0). Among all of the external nodes the one with the highest probability of failure is chosen as the most likely damage state for the structure (assume damaged state S^j which denotes a state where element j has failed), and this node is now considered as an internal node. In this damaged state (where element j has failed) the probabilities of failure for all the remaining elements are calculated which leads to a new set of external nodes. Again, among all of the existing external nodes in the failure tree the one with the highest probability is selected. This node can either belong to the newly obtained external nodes or the external nodes that were obtained formerly (so one may even have to go back to a previous state of the structure). This process continues until a collapse state is reached (a terminal node is identified). The node which corresponds to this damaged state becomes an internal node. The obtained sequence addresses the most probable failure sequence for the structure. Other important failure sequences can also be obtained by searching among the other external nodes by following the same procedure. The approach can be continued until a certain number of terminal nodes are identified.

6.4.4 β-Unzipping method

6.4.4.1 Introduction

The β-unzipping method is a method of system reliability calculation that works with the computation of reliability indices for each of the failure elements in different damaged states of the structure (or undamaged for state S^0) and finding the proper failure modes is based on those computed reliability indices. The method is quite general in a sense that it can be used for two or three dimensional trussed and framed structures [45]. In this method the whole structure is considered as a series system and the elements of this series system are parallel systems that are composed of failure elements. The procedure for system failure probability computation can be outlined as below [45]:

1. Approximating the reliability index for the elements in each parallel system.
2. Estimation of reliability index for the whole parallel system.
3. Finding an approximate equivalent linear safety margin for each one of the parallel systems.
4. Estimating the correlation between the parallel systems.
5. Finding the reliability index for the series system (structural system).

6.4.4.2 General procedure of the β-unzipping method

In the β-unzipping method the system reliability calculation is performed in different levels. It can start from level zero and go up to a level where structural collapse can happen. These levels will be explained below:

Level zero: at level zero the reliability index or failure probability is calculated for all of the failure elements in the intact structure, (a failure element can be a structural member or a cross section; however, for trusses each component is a failure element) and it is assumed that the system reliability is equal to the reliability of the element with the lowest reliability index. It will, however, give a conservative estimation of the system failure probability or reliability index:

$$\beta_S^0 = \min_{i=1,2,..n} \beta_i \tag{6.2}$$

Level one: at this level, it is assumed that failure of a single failure element can cause the failure of the whole structural system. For instance, in a truss structure failure of one of the members can cause the failure of the whole truss or for a framed structure formation of a plastic hinge can cause the failure of the whole frame. As a result, at this level the failure probability (reliability index) of the whole system is equal to the failure probability of the series system consisting of failure elements (for instance for a truss structure a set of members can form the series system). The failure is calculated according to the correlation between the safety margins of the failure element forming the series system. Let there be n failure elements in the series system as shown in Figure 6.6,then the set of elements can be chosen based on the calculated reliability indices of the failure elements for the intact structure. For instance, the components with high probabilities of failure (or low reliability indices) can be selected to form series system.

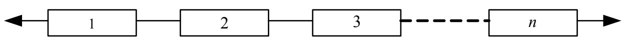

Figure 6.6: Series system at level 1

The system reliability at level one cannot be a very accurate estimation of the system reliability if the structure is not statically determinate, and it is due to the fact that indeterminate structures possess some degree of redundancy. Therefore, if one element fails there will be a redistribution of forces and the whole structural system might not fail.

Level two: at level two the system is modelled again as a series system, but each element of this series system consists of two parallel failure elements. In this level in order to create the pair of parallel elements, the most critical failure element is removed from the structure and replaced by fictitious loads corresponding to its capacity if the element is ductile (if it's brittle no load will be added since the post-failure capacity is zero). As a result, the structure will enter a new state, say S^i, where failure element i has failed. At this new state, the structure is again reanalysed and reliability computation is performed for the rest of the failure elements in the damaged state of the structure. Having the reliability indices of failure elements available at this state, pairs of failure elements can be formed with failure element i. System modelling at level 2 can be shown schematically as below:

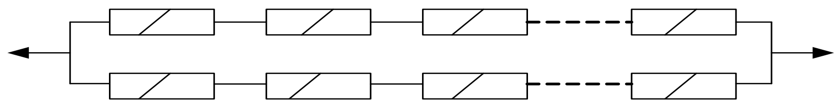

Figure 6.7: System modelling at level 2

The skewed lines on the element show that the failure of the element is ductile.

Higher levels: system reliability calculation can be done at higher levels as well (depending on the redundancy of the structure under investigation). For instance, at level three the whole system is modelled as series system where elements of the series system consist of three parallel failure elements. In general, if the structure is assumed to be failed when a mechanism is formed, system reliability calculation at that level can be called system reliability analysis at mechanism level [45].

6.4.4.3 System reliability calculation for different levels:

In the previous section the general procedure of system reliability evaluation using the β-unzipping method was explained. In this section the procedure is explained in detail. In the β-unzipping method the assumption is that all the basic random variables are normally distributed. However, this method can also be used for the case where non-normal basic random variables exist. In such cases an iterative method like FORM can be implemented to get the reliability indices for failure elements. If there are n basic random variables involved, the vector of standardized normally distributed basic random variables can be shown as $\bar{Y} = \{Y_1, Y_2, ..., Y_n\}$, and φ_n can be considered as the joint probability density function. Lets say the performance function (limit state function) for a failure element is shown by $f : \omega \rightarrow R$, then the failure region can be considered as the region where the value of failure function is smaller or equal to zero for each realisation of the random variables. The probability of failure is, therefore, calculated as below [45]:

$$P_f = P(f(\bar{Y}) \leq 0) = \int\limits_{\omega_f} \varphi_n(\bar{y}) d\bar{y} \tag{6.3}$$

In Equation 6.3 ω_f is the failure surface over which the integration is performed. The failure function then can be linearised as shown in Equation 6.4 assuming that β is the distance from the design point to the origin of the coordinates system. Equation 6.4 gives and approximation for the probability of failure:

$$P_f = P(\alpha_1 Y_1 + \alpha_2 Y_2 + ... + \alpha_n Y_n \leq -\beta) = \Phi(-\beta) \tag{6.4}$$

In the above-mentioned equation $\bar{\alpha} = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ is the vector of directional cosines and β is the Hasofer-Lind reliability index (or the reliability index from FORM where the random variables are not normally distributed). Consequently, the linearised safety margin in the standardised independent normal space can be shown as:

$$g_i(Y) = \beta_i + \sum_{j=1}^{n} \alpha_{ij} Y_j \tag{6.5}$$

In Equation 6.5 i refers to the i^{th} element and j^{th} random input variable. As noted earlier, in this approach the system is modelled as a series system. And for a series system the failure is defined as the failure of any of the elements. Thus, the failure of the system is the union of the failure events:

$$P_{f_{sys}} = P(\bigcup_{i=1}^{k} g_i(Y)) = P(\bigcup_{i=1}^{k} \bar{\alpha}_i \bar{Y} \leq -\beta) = 1 - P(\bigcap_{i=1}^{k} -\alpha_i \bar{Y} < \beta_i) = 1 - \Phi_k(\bar{\beta}, \bar{\rho}) \tag{6.6}$$

For parallel systems the probability of failure of the system can be calculated as shown in Equation 6.7

$$P_{f_{sys}} = P(\bigcap_{i}^{k} g_i(Y)) = P(\bigcap_{i}^{k} (-\bar{\alpha}_i \bar{Y} < -\beta_i) = \Phi_k(\bar{\beta}, \bar{\rho}) \tag{6.7}$$

In order to better understand the equations above, first it should be noted that the expression in Equation 6.7 ($\bigcap\limits_{i=1}^{k} (-\bar{\alpha}_i \bar{Y} < \beta_i)$) is the intersection of k linearised limit states or failure functions of the failure elements in the series or parallel system (it is presumed that there are k critical failure elements). For each one of these k failure functions there is a vector of directional cosines $\bar{\alpha}_i = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ that is multiplied by the vector of basic random variables in the normal standardised Y space. $\bar{\beta}$ is the vector containing the reliability index values for each one of the k critical failure functions, so $\bar{\beta} = \{\beta_1, \beta_2, ..., \beta_k\}$. It is also important to calculate the correlation between the failure elements in the system. The correlation matrix can be obtained by $\rho_{ij} = \bar{\alpha}_i^T \bar{\alpha}_j$ for all $i \neq j$.

6.4.4.4 Evaluation of system probability and formation of failure modes

Level 1: At level 1 the whole structural system is modelled as a series system (Figure 6.6). If there are in total n failure elements in a structure (for instance, a truss structure with n components), at level zero the reliability index has to be calculated for each of the failure elements. This can be easily done using any of the methods explained in previous chapters. However, it is not necessary to use all of these n elements to estimate the system reliability at this level. In fact, the failure probability of a series system with n elements can be calculated with sufficient accuracy by only including some of the failure elements [45]. These important elements can be selected by considering only the elements that are in the range $[\beta_{\min}, \beta_{\min} + \Delta\beta_1]$ and $\Delta\beta_1$ can be an arbitrarily chosen positive number (the bigger $\Delta\beta_1$ is, the more accurate the result). The selected set of elements are called critical elements. Knowing the critical elements, Equation 6.5 has to be evaluated in order to get the system reliability at level 1.

Equation 6.6 cannot be evaluated directly, but there are some upper and lower bounds available that can be utilized to at least get the bounds for this equation. These bounds are called Ditlevsen Bounds and are shown below [10]:

$$\max_{i=1}^{n}\left(\Phi(-\beta_i)\right) \leq P_f \leq 1 - \prod_{i=1}^{n} \Phi(\beta_i) \tag{6.8}$$

Equation 6.8 gives the bounds for a series system where its elements are either fully correlated (lower bound) or uncorrelated (upper bound). These bounds are also known as first-order series bounds. If the elements of the series system are equally correlated the following equation can be used [11]:

$$P_f = 1 - \int_{-\infty}^{+\infty} \varphi(t) \prod_{i=1}^{n} \Phi\left(\frac{\beta_i - \sqrt{\rho}t}{\sqrt{1-\rho}}\right) dt \tag{6.9}$$

Also if the correlation coefficients are not equal, the average correlation coefficient can be used in Equation 6.9 in order to get an approximation of the probability of failure. The average correlation coefficient can be calculated as below:

$$\bar{\rho} = \frac{1}{n(n-1)} \sum_{i,j=1}^{n} \rho_{ij} \tag{6.10}$$

In Equation 6.10 $i \neq j$.

There are some tables available in the literature [37] that give the values of failure probability from Equation 6.8 for different values of β, ρ, and n.

Level 2: at level 2 each element of the series system consists of a parallel pair of failure elements (critical pairs of failure elements). If there are k critical failure elements at level 1 and element h has the lowest reliability index among those k elements, then element h has to be removed from the structure (element h is assumed to have failed). if element h is ductile, it is replaced by a pair of fictitious loads representing its post-failure capacity. These fictitious loads are stochastic variables, and are equal to the post-failure resistance of failure element h depending on the assumed material behaviour of the element (refer to Figures 6.1 to 6.3). Now the structure in damaged state S^h is analysed under the external load vector $F = [F_1, F_2, ..., F_k]$ and the factitious load f_h. As a result, the load effect in the elements of the structure given element h has failed will be:

$$E_{i|h} = \sum_{j=1}^{k} a_{ij} F_j + \dot{a}_{ih} F_h \tag{6.11}$$

In Equation 6.11 a_{ij} is the so-called influence coefficient with respect to the external load vector F and $\dot{a}_{ih}$ is the influence coefficient which corresponds to f_h, the fictitious load that is replaced in lieu of member h. The safety margin for member is then as below:

$$G_{i|h} = \min(R_i^+ - E_{i|h}, R_i^- - E_{i|h}) \tag{6.12}$$

R_i^+ and R_i^- are the resistances of member in tension and compression respectively. Using the methods described in previous chapters the reliability indices can be calculated for all the remaining elements in the structure for state S^h. Among all the calculated reliability indices the smallest one is chosen ($\beta_{\min}$) and a range such as $[\beta_{\min}, \beta_{\min} + \Delta\beta_2]$ can be defined where $\Delta\beta_2$ is an arbitrarily chosen positive value. Failure elements that have reliability indices within this range are then combined with element h to form the critical failure pairs for the parallel systems.

Knowing the limit state functions for element h (G_h), which was obtained at level one and the limit state function for other elements given failure in element h ($G_{i|h}$), it is possible to obtain the probability of failure for the parallel system that consists of these two safety margins.

$$P_f = \Phi_2(-\beta_1, -\beta_2, \rho) \tag{6.13}$$

In Equation 6.13, Φ_2 is the cumulative function of a standardised bivariate normal distribution and ρ is the correlation coefficient between the two safety margins G_h and $G_{i|h}$ which is calculated as explained in Section 6.4.4.3.

The joint cumulative distribution function of Equation 6.13 has the form demonstrated below [32]:

$$F_X(X, \rho) \equiv P\left[\bigcap_{i=1}^{2} (X_i \leq x_i)\right] \equiv \int_{-\infty}^{x_1} \int_{-\infty}^{x_2} f_x(u, v, \rho) du dv \tag{6.14}$$

And in the standardized space the random variables can be shown:

$$y_i = \frac{x_i - \mu_{X_i}}{\sigma_{X_i}} \tag{6.15}$$

Therefore the joint cumulative distribution function will be expressed:

$$F_X(x, \rho) = \Phi_2(y, \rho) \tag{6.16}$$

Comparing equations 6.16 and 6.13 yields:

$$y = [-\beta_1, -\beta_2] \tag{6.17}$$

The vector of mean values and the covariance matrix in standardised Y space are:

$$\mu_{Y_i} = [0, 0] \tag{6.18}$$

$$[C] = \begin{bmatrix} CoV(Y_1, Y_1) & CoV(Y_1, Y_2) \\ CoV(Y_2, Y_1) & CoV(Y_2, Y_2) \end{bmatrix} \tag{6.19}$$

So the correlation matrix is:

$$[\rho] = \begin{bmatrix} 1 & \rho_{12} \\ \rho_{21} & 1 \end{bmatrix} \tag{6.20}$$

In order to calculate P_f the integral of Equation 6.14 needs to be solved. It can be achieved by using a technical computation software like Matlab ("mvncd" function). Another way of getting the probability of failure is using the bound defined for bivariate normal distributions [32]:

$$\max\{\Phi(-b)\Phi(-h), \Phi(-a)\Phi(-k) \leq \Phi_2(-h, -k, \rho) \leq \Phi(-b)\Phi(-h) + \Phi(-a)\Phi(-k) \tag{6.21}$$

Where:

$$h = \frac{x_1 - \mu_{X_1}}{\sigma_{X_1}} \tag{6.22}$$

$$k = \frac{x_2 - \mu_{X_2}}{\sigma_{X_2}} \tag{6.23}$$

$$a = \frac{h - \rho k}{\sqrt{(1 - \rho^2)}} \tag{6.24}$$

$$b = \frac{k - \rho h}{\sqrt{(1 - \rho^2)}} \tag{6.25}$$

For Equation 6.13, $\beta_1 = h$, $\beta_2 = k$, and $\rho = \rho_{12}$ which is the correlation between the two safety margins. The bound introduced in Equation 6.21 could be difficult to use if it is not small enough; therefore, preferably Equation 6.13 should be evaluated using technical mathematical calculation software packages such as Matlab.

Once the probability of failure for each pair of critical elements is calculated the aforementioned Ditlevsen bounds can be used for the whole system.

Level 3: the same concepts are applicable in level three only the series system comprises of elements that are composed of three parallel elements. In other terms, the series system is composed of critical triples of failure elements.

Let the critical pair of elements h and q be the pair with the lowest reliability index in the reliability evaluation at level 2, then both of these failure elements are removed from the structure and replaced by their corresponding resistances as a pair of fictitious loads (F_q and F_h). The load effect in remaining elements can be written as:

$$E_{i|h,q} = \sum_{j=1}^{k} a_{ij} F_j + {a'}_{i,h} f_h + {a'}_{i,q} f_q \tag{6.26}$$

The safety margin will be:

$$G_{i|h,q} = \min(R_i^+ - E_{i|h,q}, R_i^- + E_{i|h,q}) \tag{6.27}$$

Now that elements h and q are removed from the structure, the safety margin can be defined for each one of the remaining elements. The next step is to calculate the reliability index for all of these remaining failure elements. If the minimum reliability index is $\beta_{\min}$, then failure elements with reliability values in the range $[\beta_{\min}, \beta_{\min}+\Delta\beta_3]$ can be utilized to form the parallel triples of failure elements. Again, $\Delta\beta_3$ is an arbitrarily chosen positive number. The probability of failure can be calculated for each parallel system by considering the three dimensional standardised normal cumulative distribution function of Equation 6.28.

$$P_f = \Phi_3(-\beta_1, -\beta_2, -\beta_3, \bar{\rho}) \tag{6.28}$$

These three reliability indices are the ones calculated at levels 1, 2, and 3. If elements h, q, and p are the failure elements forming a parallel system, then β_1 is the reliability index corresponding to safety margin G_h, β_2 is the reliability index corresponding to $G_{q|h}$, and finally β_3 is the reliability index corresponding to $G_{p|h,q}$. And $\bar{\rho}$ is the correlation between these safety margins. Equation 6.28 can be evaluated the same way by using a technical mathematical calculation software such as Matlab.

The calculation can be continued to higher levels in the same manner. Nonetheless, a system reliability calculation up to level 3 seems sufficient for most truss structures. In fact, evaluation at higher level needs highly redundant structures so that the system collapse is not occurring at higher levels.

6.4.4.5 Calculation of the equivalent safety margins for parallel systems

It was mentioned in previous sections that in system reliability analysis using the β-unzipping method the whole system is modelled as a series system. From level two onwards, each element of the series system is itself a parallel system that is composed of some elements. In Section 6.4.4.4, it was explained how to calculate the system reliability of the parallel system. However, an equivalent linear safety margin is also required for each parallel system. These equivalent safety margins can then be used to calculate the correlation matrix for the whole structural system (series system). The procedure to obtain the equivalent safety margins is as follows [18, 36]: If the parallel system is composed of n failure elements, then by using Equation 6.7 the reliability index of the system can be computed as shown in Equation 6.29.

$$\beta_P = -\Phi^{-1}\left(\Phi_n(-\bar{\beta}; \bar{\rho}\right) \tag{6.29}$$

Where $\bar{\beta}$ is the vector of reliability index values of the individual elements composing the parallel system and $\bar{\bar{\rho}}$ is the vector of correlations between the elements of the parallel system.

The equivalent safety margin is then calculated in a way that β_{eq} is equal to β_P where the equivalent safety margin (G^e) possesses the same sensitivity with respect to changes in the basic input random variables as the parallel system itself [36]. Consequently, if the vector of basic input random variables ($\bar{Y}$) is changed by a vector $\bar{\varepsilon} = [\varepsilon_1, \varepsilon_2, ..., \varepsilon_k]$, then the reliability index of the parallel system with respect to the change is:

$$\beta_P(\bar{\varepsilon}) = -\Phi^{-1}\left(P\left(\bigcap_{i=1}^{n}\left\{\sum_{j=1}^{k}\alpha_{ij}(Y_j+\varepsilon_j)+\beta_i \leq 0)\right\}\right)\right) = -\Phi^{-1}(\Phi_n(-\bar{\beta}-\bar{\alpha}\times\bar{\varepsilon};\bar{\rho})) \quad (6.30)$$

Based on the value of β_P obtained through Equation 6.30 and if the equivalent linear safety margin for the parallel system (G^e) is as shown in Equation 6.30, then:

$$G^e = \alpha_1^e Y_1 + \alpha_2^e Y_2 + ... + \alpha_k^e Y_k + \beta^e \quad (6.31)$$

In Equation 6.30, $\bar{\alpha}^e = [\alpha_1^e, \alpha_2^e, ..., \alpha_k^e]$ the vector of equivalent sensitivity factors and $\beta^e = \beta_P$. If the same perturbation (change) is made to the basic input random variables, then the reliability index is obtained as below:

$$\beta_P(\bar{\varepsilon}) = -\Phi^{-1}(\Phi(-\beta^e - \bar{\alpha}^{eT}\bar{\varepsilon})) = \beta^e + \alpha_1^e\varepsilon_1 + \alpha_2^e\varepsilon_2 + \cdots + \alpha_k^e\varepsilon_k \quad (6.32)$$

Using Equation 6.32 and if $\bar{\varepsilon} = 0$ [36]:

$$\alpha_i^e = \frac{\left.\frac{\partial\beta_P}{\partial\varepsilon_i}\right|_{\bar{\varepsilon}=\bar{0}}}{\sqrt{\sum_{j=1}^{n}\left(\left.\frac{\partial\beta_P}{\partial\varepsilon_i}\right|_{\bar{\varepsilon}=\bar{0}}\right)}}, i = 1, 2, \ldots, k \quad (6.33)$$

Equations 6.29 to 6.33 can be used for obtaining the equivalent safety margins; however, these equation are rather complex. A numerical differentiation method such as finite difference method can be used for the computation of the equivalent safety margins. The following steps can be followed for this purpose.

1. Calculate the reliability index for the parallel system (β_P) using Equation 6.29.
2. Perturb the value of basic random variable (Y_i) by a small amount while keeping the rest of the random variables at their original value, so the vector of perturbation has the form $\bar{\varepsilon} = [0, 0, \ldots, \varepsilon_i, \ldots, 0]$.
3. obtain the value of $\bar{\beta}_i = -\bar{\beta} - \bar{\rho} \times \bar{\varepsilon}^T$ where $\bar{\beta}$ is the vector containing the reliability indices of the elements of the parallel system and $\bar{\rho}$ is the correlation matrix of the parallel system, and $\bar{\varepsilon}$ is the the vector obtained in step 2.

4. Calculate the value of β_{P_i} using Equation 6.34.

$$\beta_{P_i} = -\Phi^{-1}(\Phi_n(-\bar{\beta}_i, \bar{\rho})) \tag{6.34}$$

5. Obtain the value of the non-normalised equivalent sensitivity factors as in Equation 6.35.

$$\alpha_i = \frac{\beta_{P_i} - \beta_P}{\varepsilon_i} \tag{6.35}$$

6. Repeat steps 2 to 5 for the rest of the basic input random variables.

7. Normalise the obtained sensitivity factors using Equation 6.36.

$$\alpha_{iN} = \frac{\alpha_i}{\sqrt{\sum_{i=1}^{k} \alpha_i^2}} \tag{6.36}$$

8. Form the equivalent safety margin as below.

$$G^e = \alpha_{N1}^e Y_1 + \alpha_{N2}^e Y_2 + \cdots + \alpha_{Nk}^e Y_k + \beta_P \tag{6.37}$$

6.5 Conclusion on background to system reliability

Throughout this chapter two main methods of system reliability were explained. The β-unzipping method was discussed comprehensively and the procedure for the branch and bound method was also explained. The formulations that were explained for the β-unzipping method can also be used for the branch and bound method. The only difference between the two is that they follow different search criteria to form the most important failure sequences. In the β-unzipping method the identification of the failure modes (failure sequences) is based on the determination of an unzipping interval which is based on the reliability index values obtained at different levels. On the other hand, in the branch and bound method, the formation of the failure sequences is based on the formation of the failure tree for different damaged states of the structure and inspection of the results where the external nodes that are most likely to occur are used for the identification of the failure paths. In Chapter 7, these two methods are used for the system reliability evaluation of a structure where computer algorithms are developed for the system reliability calculation.

Chapter 7

System Reliability Evaluation of Truss Structures

7.1 Introduction

In Chapter 6 different methods for the system reliability evaluation of truss structures were discussed. In this chapter the system reliability analysis of a 10-bar truss structure is investigated using the β-unzipping method of system reliability analysis as well as the branch and bound search method. This way it is possible to evaluate the applicability of the methods for system reliability evaluation of truss structures. Finally, the computerised methodology of system reliability evaluation will be discussed. The 10-bar truss structure considered in this chapter is the same as the one considered in Chapter 5 of the thesis. Therefore, the resistance of all members are assumed to be in terms of yield stress (f_y) with the mean value of 407 and standard deviation of 28.5 and both loads (F_1 and F_2) are in terms of (F) with the mean value of 174 and standard deviation of 60.9 ($F_1 = F$ & $F_2 = 0.8F$) [24, 27, 43]. Also, both random variables are assumed to be mutually independent. The evaluation results are presented in the following sections. For convenience the structure is shown again.

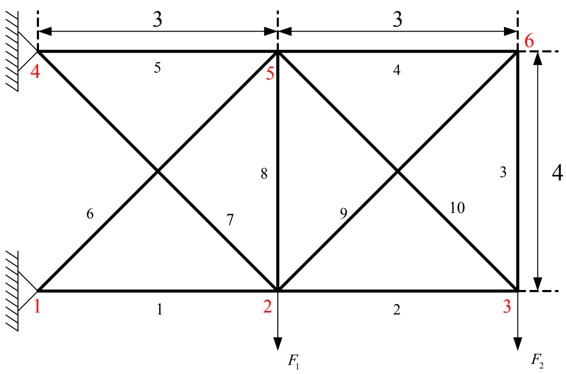

Figure 7.1: 10-bar truss structure

7.2 System reliability analysis using the β-unzipping method

As was previously mentioned in Chapter 6 that when using the β-unzipping method, analysis is performed at different levels. Two cases need to be considered: the first case is when all the basic random variables are normally distributed, the second case is when all the input basic random variables are non-normal. The case with normal random variables is assessed first since it is easier to perform, and it helps establish the concepts and procedures better. The case of non-normal random variables is more complicated, and needs to be performed along with an iterative reliability analysis method such as the FORM method. This case is investigated after the concepts and procedures are established using the simpler case where all the variables are normal. It should also be noted that it is assumed that the failure of all the elements of the structure is ductile. The system reliability evaluation results for different levels of system reliability analysis for both of the cases are shown in the following sections.

7.2.1 System reliability analysis - Normal random variables

system reliability analysis of the structure is performed where all the input basic random variables are normally distributed. This is shown in Sections 7.2.1.1 to 7.2.1.4.

7.2.1.1 System reliability analysis at level zero

At level zero the system reliability is assumed to be equal to the reliability of a component with the lowest reliability index. This means that only a component-level reliability analysis of the intact structure is required. A component-level reliability evaluation of the truss structure shown in Figure 7.1 was performed in Chapter 5. These values are shown in Table 7.1 below.

Table 7.1: Reliability indices of the truss elements at level zero

Member	β	P_f
1	3.2634	0.00055
2	3.2157	0.00065
3	5.5295	1.61E-08
4	7.3373	1.09E-13
5	3.2639	0.00055
6	3.5298	0.00021
7	3.0797	0.00104
8	5.5759	1.23E-08
9	5.2670	6.93E-08
10	3.0960	0.00098
Minimum RI	3.0797	0.00104

As can be seen from the table the lowest reliability index belongs to member 7. Therefore, the reliability of the system at level zero is as below.

$$\beta_s^0 = \min \beta_i = \beta_7 = 3.0797 \tag{7.1}$$

7.2.1.2 System reliability analysis at level one:

At level one the reliability of the system is performed assuming that the components with the reliability values in the interval $[\beta_{\min}, \beta_{\min} + \Delta\beta]$ will form a series system where the failure of any of the components will result in the failure of the whole structural system.

Here, it is assumed that $\Delta\beta = 1$. As a result, the interval will be $[3.0797, 4.0797]$ which means member 1,2,5,6,7, and 10 can be considered as the critical elements forming the series system.

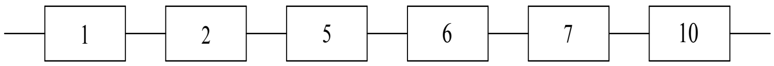

Figure 7.2: System reliability model at level 1

In order to calculate the system reliability two methods can be applied. The first method is using Ditlevsen Bounds (Equation 6.8). The lower Ditlevsen bound corresponds to the case where all the elements are fully correlated and the upper bound corresponds to the case where all the elements are uncorrelated. For the structure the Ditlevsen bound is as shown:

$$0.001036 < P_f < 0.003969 \tag{7.2}$$

Another method to calculate the reliability of the series system is to use Equation 6.6 for the series system. This will require the calculation of the correlation matrix for the series system. Computation of correlation matrix in turn needs the calculation of sensitivity factors for the random variables in each one of the component's safety margins (limit state functions). These computations are performed using the FOSM reliability method since all of the random variables are normal. If the limit state has the form $G = a \times R - b \times E$, then the reliability index, and sensitivity factors with respect to the resistance and load effect are calculated as below assuming that the random variables are not correlated:

$$\beta = \frac{a\mu_R - b\mu_E}{\sqrt{(a\sigma_R)^2 + (b\sigma_E)^2}} \tag{7.3}$$

$$\alpha_R = \frac{a\sigma_R}{\sqrt{(a\sigma_R)^2 + (b\sigma_E)^2}} \tag{7.4}$$

$$\alpha_E = \frac{-b\sigma_E}{\sqrt{(a\sigma_R)^2 + (b\sigma_E)^2}} \tag{7.5}$$

Once the value of the sensitivity factors are calculated for each limit state functions, it is possible to compute the correlation matrix using Equation 7.6

$$\rho_{ij} = \bar{\alpha}_i^T \bar{\alpha}_j \text{ for all } i \neq j \tag{7.6}$$

Where $\bar{\alpha}_i = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ is the vector of directional cosines (sensitivity factors) for each safety margin in the system. Using the aforementioned equations, the sensitivity factors were calculated.

It should be noted that the reliability indices were already calculated (Table 7.1). In Table 7.2 the factors of resistance and load effect (a,b) in the limit state of each of the failure elements are given as well as the corresponding sensitivity factors.

Table 7.2: Element sensitivity factors at level 1

Member	a	b	α_R	α_E
1	1.2293	1.2767	0.41082	0.91171
2	0.4778	0.5005	0.40790	0.91303
5	1.2260	1.2733	0.41082	0.91172
6	1.1352	1.1222	0.42788	0.90384
7	1.0490	1.1278	0.39911	0.91690
10	0.7783	0.8342	0.40014	0.91645

The sensitivity factors were used in Equation 7.6 to obtain the correlation coefficients matrix.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

It can be seen from the obtained correlation matrix that the safety margins of the failure elements of the series system of Figure 7.1 are fully correlated. Therefore, the lower Ditlevsen bound is the best estimate of failure probability for the series system. Therefore, the failure probability of the system at level one can be calculated as below:

$$P^1_{f_{sys}} = 0.001036$$

And the reliability index for the system at level one is:

$$\beta^1_S = -\Phi^{-1}(0.001036) = 3.0797$$

The correlation coefficients matrix shows full correlation between the safety margins of the failure elements forming the series system since the same resistance and load effect is used for all of the elements. In other terms, since the basic input random variable for the resistance of all members is assumed to be the same yield stress random variable and the load effect of all of the elements is in terms of one applied load variable a high correlation is created between the safety margins of the structural members.

7.2.1.3 System reliability analysis at level two:

At level two the system reliability is calculated where each element of the series system is a parallel system of two failure elements on its own. At this level, the most critical elements (element with the reliability indices within the unzipping interval) are removed from the structure one at a time, and are replaced with a set of fictitious loads representing their post-failure capacities (in the case of ductile or semi-ductile failure). This procedure can be outlined algorithmically as below:

1. Analyse the intact structure and calculate the reliability indices for all the components (failure elements) of the structure.
2. Define a set of critical elements by considering the elements with the reliability values within the interval $[\beta_{\min}, \beta_{\min} + \Delta\beta_1]$ where $\Delta\beta_1$ is a value that is chosen arbitrarily, and $\beta_{\min}$ is the smallest reliability index of all the failure elements in the intact structure.
3. Remove the most critical failure element from the structure, and replace it with fictitious loads equal to its post-failure capacity (if the failure is brittle no fictitious loads are added).
4. Perform a stochastic finite element analysis of the damaged structure, and calculate the reliability indices for the structure at its new damaged state.
5. Define a second set of critical failure elements for the structure. The elements with reliability indices within the interval $[\beta_{\min}, \beta_{\min} + \Delta\beta_2]$ are considered as critical failure elements where $\Delta\beta_2$ is selected arbitrarily and $\beta_{\min}$ is the smallest reliability index of all the remaining failure elements in the new damaged state of the structure.
6. Pair the critical failure elements from step 5 with the failure elements selected in step 2 to form a set of parallel systems.
7. Repeat steps 3 to 6 for the rest of the critical failure elements determined in step 2 in order to obtain more critical pairs of failure elements.
8. For each one of the critical pairs of failure elements obtain an estimation of the failure probability and an equivalent safety margin.
9. Obtain the correlation between the estimated safety margins of the critical pairs of failure elements (forming the matrix of correlation coefficients for the elements of the series system).
10. Calculate the system reliability of the series system using the information obtained in steps 8 and 9 (forming the Ditlevsen bounds).

The first step in calculation of the system reliability of the ten bar truss structure at level 2 is to get the reliability indices of the failure elements in the undamaged (intact) state of the structure which was already performed in Chapter 5.

The next step is to determine a set of critical failure elements. This set can be the same as the ones determined in level one. However, here a smaller interval is selected to decrease the number of failure element pairs that have to be considered. It is clear that a bigger interval will lead to a more accurate estimate of the system reliability, yet making the calculations more time-consuming.

If $\Delta\beta_1 = 0.15$, then the interval to determine the critical failure elements is $[3.0797, 3.2297]$ where $\beta_{\min} = 3.0797$. According to Table 7-1 members 7, 2, and 10 have to be considered as the critical failure elements since their reliability indices fall within the β-unzipping interval.

Removing member 7: Member 7 with the lowest reliability index is removed from the structure as the first element to consider. Member 7 fails in tension and is replaced by fictitious loads in that it is assumed that the failure is ductile. This is shown in Figure 7.3 below. This damaged state of the structure is denoted with (S^7).

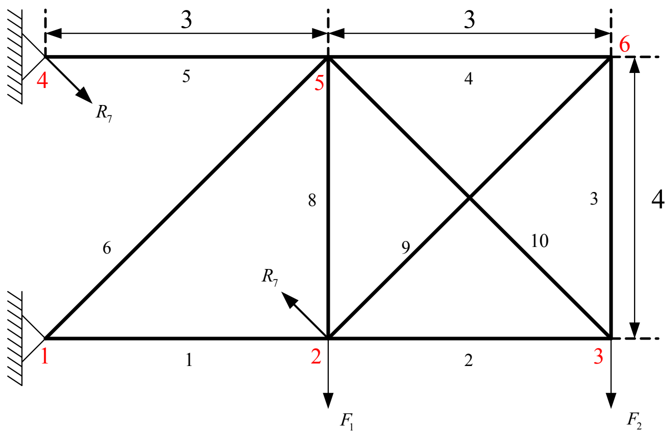

Figure 7.3: 10-bar truss structure with ductile failure of member 7

After element 7 is removed from the structure, the new structure needs to be analysed elastically in the new state where the fictitious loads representing the post-failure capacity of member 7 are applied to the structure. Through the elastic finite element analysis of the structure, it is possible to obtain the influence factors with respect to the original loads as well as the fictitious loads with which member 7 was replaced. To obtain the influence factors each time a unit load is applied in the direction of the applied loads one at a time (once a unit load is applied in the direction of R_7 and once in the direction of $F_1 = F = 1$ and $F_2 = 0.8F = 0.8$). In other terms, influence factors (influence coefficients) with respect to a certain load are internal member forces where a unit load is applied in the direction of that load.

Influence factors with respect to the applied loads for the structure in the damaged state where member 7 is removed are shown in Table 7.3.

Table 7.3: Influence factors with respect to a_F and a_{R_7}-member 7 in failure state

Member	a_F	a_{R_7}
1	-0.6000	-0.6294
2	-0.6311	0.1215
3	-0.0415	0.1620
4	-0.0311	0.1215
5	1.9500	-0.6294
6	-2.2500	1.0490
8	0.9585	-0.6772
9	0.0519	-0.2025
10	1.0519	-0.2025

The formation of the safety margins in the new state is as follows: The load effect in each member (i) is determined using Equation 7.7:

$$E_{i|h} = a_{F_i} F + a_{R_7} R_7 \tag{7.7}$$

As is clear from Equation 7.7 and Table 7.3, for some of the members the influence factors have different signs (for instance, member 2). It means the members can either be in tension or compression, and since the ratios of the influence coefficients are also random in nature, the member can either fail in tension or compression. As a result, the safety margin (limit sate equation) can be expressed as below:

$$G_{i|7} = \min(R_i^+ - E_{i|7}, R_i^- - E_{i|7}) \tag{7.8}$$

According to Equation 7.8, the reliability calculations are performed with respect to both failure in compression and failure in tension. It is, however, important to note the fact that failure of some of the members in a certain mode will be impossible and the calculation of reliability indices with respect to that failure mode would yield irrational results or cause the divergence of the reliability calculation algorithm in the case of non-normal random variables. Therefore, it is important to check the possibility of failure in a certain mode for all of the members.

Resistances of the remaining members are shown in Table 7.4 below for both tension and compression failure modes.

Table 7.4: Resistance factors for compression (R_C) and tension (R_T)

Member	R_C	R_T
1	1.2296	1.9164
2	0.4778	0.9293
3	0.0074	0.1885
4	0.0131	0.1885
5	0.6232	1.2265
6	1.1356	2.2704
8	0.0177	0.3299
9	0.2258	0.9293
10	0.1907	0.7783

In Table 7.4, the resistance factors are shown. These are the factors that are multiplied by the yield strength of the members to obtain their resistances.

In Table 7.5, the limit state functions for both tension and compression failure of the remaining failure elements are shown.

Table 7.5: Safety margins for failure in compression($G^-_{i|7}$) and tension ($G^+_{i|7}$)-member 7 in failure state)

	Safety Margins	
Member	$G^-_{i\|7}$	$G^+_{i\|7}$
1	$G^-_{1\|7} = 1.2296f_y + (-0.6000F - 0.6294f_y)$	$G^+_{1\|7} = 1.9164f_y - (-0.6000F - 0.6294f_y)$
2	$G^-_{2\|7} = 0.4778f_y + (-0.6311F + 0.1215f_y)$	$G^+_{2\|7} = 0.9293f_y - (-0.6311F + 0.1215f_y)$
3	$G^-_{3\|7} = 0.0074f_y + (-0.0415F + 0.1620f_y)$	$G^+_{3\|7} = 0.1885f_y - (-0.0415F + 0.1620f_y)$
4	$G^-_{4\|7} = 0.0131f_y + (-0.0311F + 0.1215f_y)$	$G^+_{4\|7} = 0.1885f_y - (-0.0311F + 0.1215f_y)$
5	$G^-_{5\|7} = 0.6232f_y + (+1.9500F - 0.6294f_y)$	$G^+_{5\|7} = 1.2265f_y - (+1.9500F - 0.6294f_y)$
6	$G^-_{6\|7} = 1.1356f_y + (-2.2500F + 1.0490f_y)$	$G^+_{6\|7} = 2.2704f_y - (-2.2500F + 1.0490f_y)$
8	$G^-_{8\|7} = 0.0177f_y + (+0.9585F - 0.6772f_y)$	$G^+_{8\|7} = 0.3299f_y - (+0.9585F - 0.6772f_y)$
9	$G^-_{9\|7} = 0.2258f_y + (+0.0519F - 0.2025f_y)$	$G^+_{9\|7} = 0.9293f_y - (+0.0519F - 0.2025f_y)$
10	$G^-_{10\|7} = 0.1907f_y + (+1.0519F - 0.2025f_y)$	$G^+_{10\|7} = 0.7783f_y - (+1.0519F - 0.2025f_y)$

With the limit state equations known, the next step is to calculate the reliability index and sensitivity factors for all of the 18 limit state equations. This can be done by utilising Equations 7.3 to 7.5, as was done for the calculation of reliability indices at level 1. However, it is essential to check if a certain failure mode would be applicable for a member before the calculation of the reliability index and sensitivity factors. If only one failure mode is possible for a member, then the calculations are done for that specific failure mode, Otherwise the calculations are performed for both of the failure modes (tension and compression) and the one with the smaller reliability index would be the prevailing one. For instance, it is impossible for member 8 to fail in compression. This is demonstrated below.

Consider the safety margin for member 7 in the intact structure:

$$G = 1.0490{f_y}^+ - 1.1278F \tag{7.9}$$

For the failure of member 7, it in necessary that:

$$1.0490f_y < 1.1278F \rightarrow f_y < 1.0751F \tag{7.10}$$

Now consider the safety margin for the compression failure mode of member 8 in the damaged state where member 7 has failed.

$$G^-_{8|7} = R^-_8 + (0.9585F - 0.6772f_y) \tag{7.11}$$

From Equation 7.11, it is derived that for the failure to occur:

$$R^-_8 + (0.9585F - 0.6772f_y) < 0 \tag{7.12}$$

Which means:

$$R_8^- + 0.9585F < 0.6772f_y \tag{7.13}$$

From Equation 7.13, it is concluded that:

$$1.4154F < f_y \tag{7.14}$$

Comparing 7.14 and 7.10 yields:

$$\{G_7 < 0 \cap G_{8|7}^- < 0\} = \varnothing \tag{7.15}$$

From Equation 7.15, it is evident that it is impossible for member 8 to fail in compression in the damaged state where member 7 is removed from the structure. The same procedure is performed for all of the elements to determine if a certain failure mode is not applicable. The next step is to calculate the reliability index and sensitivity factors for all of the members. The results are shown in Table 7.6.

Table 7.6: Member reliability indices-member 7 in failure state

Member	Reliability index	Governing Failure Mode
1	3.4673	Compression
2	3.1879	Compression
3	6.8272	Tension
4	11.5364	Compression
5	3.2003	Tension
6	3.3063	Compression
8	3.7373	Tension
9	5.7384	Compression
10	3.0928	Tension
Minimum RI	3.0928	Tension

As can be seen in Table 7.6, the reliability indices are calculated for the damaged state where member 7 has failed. The reliability index values shown in Table 7.6 are for the applicable failure modes. If a decision regarding $\Delta\beta_2$ is already made, it might not be necessary for certain members with high reliability index values to check the applicability of a certain failure mode. This will help decrease the calculation time.

With the minimum reliability index known (member 10, β =3.0928), $\Delta\beta_2$ is selected to be 0.2. Accordingly, the interval will be $[3.0928, 3.2928]$. It will include members 10, 2, and 5.

As a result, the following 3 pairs will be formed.The parallel-series model for these pairs is shown in Figure 7.4.

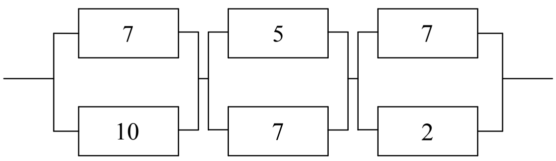

Figure 7.4: Parallel Pairs with member 7

Next step is to calculate the sensitivity factors, correlation coefficients, and the reliability index for each pair. Finally, an estimate for the safety margin of each pair will be computed, so that the total reliability of the series system can be calculated.

Sensitivity factors and correlation coefficients are calculated as shown in Table 7.7.

Table 7.7: Sensitivity factors and correlation coefficients-member 7 in failure state

Member	α_F	α_{f_y}	$\rho_{i\|7}$
2	-0.9138	0.4061	0.9999
5	-0.9134	0.4069	0.9999
10	-0.9165	0.3999	0.9999

In Table 7.7 α_F is the sensitivity factor with respect to the load effect and α_{f_y} is the sensitivity factor with respect to the resistance. These values are computed considering the prevailing safety margins of the members for the state where member 7 has failed (Table 7.7). Equations 7.4 and 7.5 are used for the calculation of the sensitivity factors. It should also be noted that for member 7, $\alpha_F = -0.9169$ and $\alpha_{f_y} = 0.3991$. These values are calculated using the safety margin of member 7 for the intact structure. The third column of Table 7.7 gives the correlation coefficients for each pair. For instance, the correlation coefficient for the parallel system of member 7 and 2 is $\rho_{2,7} = -0.9169 \times -0.9138 + 0.4061 \times 0.3991 = 0.9999$ and so forth. The correlation coefficient is calculated using Equation 7.6. With the correlation coefficients and the reliability indices known, it is possible to obtain the reliability index and probability of failure for each pair of failure elements. These values are shown in Table 7.8.

Table 7.8: RI and P_f values of the failure element pairs-member 7 in failure state

Member	β_7	$\beta_{i\|7}$	$P_{f_{pair}}$	β_{pair}
2	3.0797	3.1879	7.1655E-04	3.1879
5	3.0797	3.2003	6.8642E-04	3.2003
10	3.0797	3.0928	9.8678E-04	3.0942

In order to calculate the probability of failure Equation 6.13 is used. For example, the failure probability for the pair of elements 2 and 7 is calculated as shown below. It should be mentioned that this calculation can be readily performed using Matlab's built-in function "mvncdf".

$$P_{f_{2-7}} = \Phi_2(-3.0797, -3.1879, \rho_{2-7}) = 7.1655 \times 10^{-4}$$

After the calculation of reliability indices for each pair of failure elements the equivalent safety margins should be estimated for each pair. This is done by using the procedure explained in Chapter 6. For instance, for parallel system of members 2 and 7 the detailed calculations are as below.

The sensitivity factor for each failure element in the pair (Table 7.7):

$$\bar{\alpha}_7 = [0.3991, -0.9169]$$

$$\bar{\alpha}_2 = [0.4061, -0.9138]$$

The reliability index values for each of the failure elements and the parallel system are shown:

$$\beta_7 = 3.0797$$

$$\beta_2 = 3.1879$$

$$\beta_P = 3.1879$$

And the correlation coefficient for the pair is $\rho_{2,7} = 0.9989$.

First the sensitivity of the system with respect to the resistance can be evaluated:

$$-\bar{\beta} - \bar{\alpha}\bar{\varepsilon} = \begin{bmatrix} -3.0797 \\ -3.1879 \end{bmatrix} - \begin{bmatrix} 0.3991 & -0.9169 \\ 0.4061 & -0.9138 \end{bmatrix} \times \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} = \begin{bmatrix} -3.1196 \\ -3.2285 \end{bmatrix}$$

Where $\bar{\varepsilon} = (0.1, 0)$, since the sensitivity is evaluated with respect to the resistance variable. Now, it is possible to evaluate Equation 6.34.

$$\beta_p = -\Phi^{-1}(\Phi_2(-3.1196, -3.2285, 0.9989)) = 3.2285$$

Through numerical differentiation the sensitivity is evaluated:

$$\frac{\partial \beta}{\partial \varepsilon_1} = \frac{3.2287 - 3.1879}{0.1} = 0.4081$$

The same procedure is repeated to obtain the sensitivity with respect to the load effect as a basic random variable where $\varepsilon = (0, 0.1)$.

$$-\bar{\beta} - \bar{\alpha}\bar{\varepsilon} = \begin{bmatrix} -2.9880 \\ -3.0965 \end{bmatrix}$$

As a result:

$$\beta_p = -\Phi^{-1}(\Phi_2(-2.9880, -3.0965, 0.9989)) = 3.0965$$

And the sensitivity is:

$$\frac{\partial \beta}{\partial \varepsilon_2} = \frac{3.0965 - 3.1879}{0.1} = -0.9140$$

Finally, the sensitivity factors have to be normalised (Equation 6.36). This will lead to the following vector for the sensitivity factors which is a unit vector.

$$\bar{\alpha}^e = (\alpha_1^e, \alpha_2^e) = (0.4077, -0.9131)$$

Having calculated the equivalent sensitivity factors, it is possible to obtain the equivalent safety margin. The equivalent safety margin for the pair of elements 2 and 7 is shown below:

$$G^e_{2,7} = 0.4077f_y - 0.9131F + 3.1879$$

The same calculations are performed for the other pairs of failure elements i.e. 7,5 and 7,10. The results are tabulated below in Table 7.9.

Table 7.9: Equivalent safety margins-member 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7	0.4077	-0.9131	$G^e_{2,7} = 0.4077f_y - 0.9131F + 3.1879$
5 & 7	0.4084	-0.9128	$G^e_{5,7} = 0.4084f_y - 0.9128F + 3.2003$
10 & 7	0.4135	-0.9105	$G^e_{10,7} = 0.4135f_y - 0.9105F + 3.0942$

Removing member 2: the next step in calculation of the system reliability at level 2 is to remove member 2 from the intact structure, and perform the same procedure as above to obtain a new set of parallel systems. Member 2 fails in compression; therefore, it should be replaced by the forces shown in Figure 7.5.

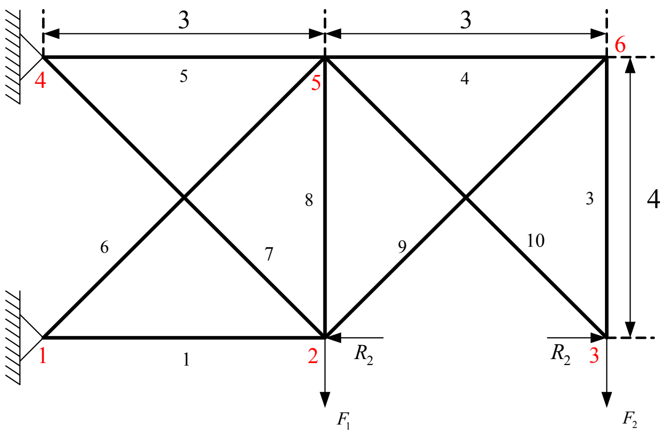

Figure 7.5: 10-bar truss structure with the ductile failure of member 2

After member 2 is removed from the structure, the structure is elastically analysed once under the external loads, and once under the fictitious loads that are equal to the post-failure resistance of member 2. The influence coefficients are presented in Table 7.10 below.

Table 7.10: Influence factors with respect to a_F and a_{R_7}-member 2 in failure state

Member	a_F	a_{R_2}
1	-1.5174	0.2290
3	0.8000	-0.6371
4	0.6000	-0.4778
5	1.0326	0.2290
6	-0.7210	-0.3817
7	1.5290	-0.3817
8	0.5768	-0.3317
9	-1.0000	0.7963
10	0.0000	0.7963

According to the influence factors the new safety margins can be formed. These safety margins are shown in Table 7.11.

Table 7.11: Safety margins for failure in compression ($G^-_{i|2}$) and tension ($G^+_{i|2}$)-member 2 in failure state

	Safety Margins	
Member	$G^-_{i\|2}$	$G^+_{i\|2}$
1	$G^-_{i\|2} = 1.2296f_y + (-1.5174F + 0.2290f_y)$	$G^+_{i\|2} = 1.9164f_y - (-1.5174F + 0.2290f_y)$
3	$G^-_{i\|2} = 0.0074f_y + (+0.8000F - 0.6371f_y)$	$G^+_{i\|2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$
4	$G^-_{i\|2} = 0.0131f_y + (+0.6000F - 0.4778f_y)$	$G^+_{i\|2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$
5	$G^-_{i\|2} = 0.6232f_y + (+1.0326F + 0.2290f_y)$	$G^+_{i\|2} = 1.2265f_y - (+1.0326F + 0.2290f_y)$
6	$G^-_{i\|2} = 1.1356f_y + (-0.7210F - 0.3817f_y)$	$G^+_{i\|2} = 2.2704f_y - (-0.7210F - 0.3817f_y)$
7	$G^-_{i\|2} = 0.3115f_y + (+1.5290F - 0.3817f_y)$	$G^+_{i\|2} = 1.0490f_y - (+1.5290F - 0.3817f_y)$
8	$G^-_{i\|2} = 0.0177f_y + (+0.5768F - 0.3317f_y)$	$G^+_{i\|2} = 0.3299f_y - (+0.5768F - 0.3317f_y)$
9	$G^-_{i\|2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	$G^+_{i\|2} = 0.9293f_y - (-1.0000F + 0.7963f_y)$
10	$G^-_{i\|2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	$G^+_{i\|2} = 0.7783f_y - (+0.0000F + 0.7963f_y)$

The reliability evaluation should be performed for all of the safety margins above. Nevertheless, some of the safety margins might not be applicable. Hence the applicable ones have to be identified the same way that was done for the state where member 7 was removed from the structure. For some of the members both of the limit state functions are applicable, and for some only one limit state is applicable. When both of the limit state functions are applicable the safety margin that yields the smaller reliability index is the one that governs. For instance, for member 1 both tension and compression failure modes are applicable; however, failure in compression yields a smaller reliability index (higher probability of failure). Thus, the compression is the governing failure mode. it is shown below.

For the failure of member 1 in compression in the undamaged state:

$$G = 0.4778f_y - 0.5005F < 0 \rightarrow f_y < 1.0475F$$

For the failure of member 1 in compression in the damaged state where member 2 has failed:

$$G^-_{1|2} = 1.2296f_y + (-1.5174F + 0.2290f_y) < 0 \rightarrow f_y < 1.0403F$$

This concept can be shown graphically where the dashed line represents the failure of member 1 when member 2 has failed, and the continuous line shows the failure of member 2 in the intact structure. The hatched area depicts the intersection of the two events. The calculated reliability index for the failure of member 1 in compression in the damaged state is calculated as 3.2532.

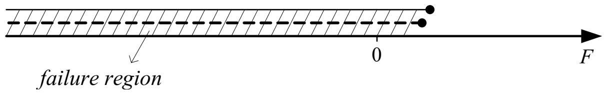

Figure 7.6: Graphical representation of compression failure for member 1

For the failure of member 1 in tension in the damaged state where member 2 has failed:

$$G^{+}_{1|2} = 1.2296f_y - (-1.5174F + 0.2290f_y) < 0 \rightarrow f_y < -0.8992F$$

This failure mode is shown graphically in Figure 7.7.

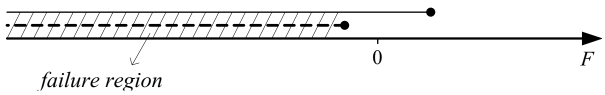

Figure 7.7: Graphical representation of tension failure for member 1

The computed reliability index for tension failure mode of member 1 is 9.1269 which is higher than the calculated reliability index for the compression failure of member 1. This fact is confirmed by the smaller failure region for tension failure mode as shown in Figure 7.7.

It is seen that for the failure of member 1 both of the failure modes are applicable. However, compression failure mode is more critical than the failure in tension, and it will be the governing limit state function.

Table 7.12: Member reliability indices-member 2 in failure state

Member	Reliability index	Governing Failure Mode
1	3.2532	Compression
3	3.6370	Tension
4	4.0501	Tension
5	3.2789	Tension
6	3.7099	Compression
7	3.1112	Tension
8	4.2362	Tension
9	3.5850	Compression
10	14.2807	Compression
Minimum RI	3.1112	Tension

Form table 7.12, it is seen that member 7 has the lowest reliability index ($\beta_{7|2} = 3.1112$). $\Delta\beta_2$ was selected to be 0.20. Therefore, the interval for the β-unzipping will be $[3.1112, 3.3112]$. From the interval, it is seen that the critical members are 1, 7, and 5. The parallel system will be as shown below:

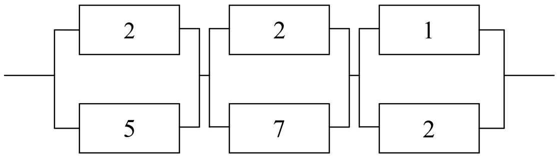

Figure 7.8: Parallel Pairs with member 2

The pair of members 2 and 7 refers to a physical damaged state of the structure that was already investigated where member 7 was removed form the structure. Nevertheless, the sequence to reach this physical state is different and it cannot be ignored. As a result, pairs of members 2-7, 2-5 and 2-1 should be investigated. The sensitivity factors and correlation coefficients are calculated for the pairs as shown in table 7.13.

Table 7.13: Sensitivity factors and correlation coefficients-member 2 in failure state

Member	α_{f_y}	α_F	$\rho_{i\|2}$
7	0.4011	-0.9160	0.9999
5	0.4119	-0.9112	0.9999
1	0.4079	-0.9130	0.9999

The sensitivity factors are calculated using the safety margins of members in the damaged state of structure where member 2 has failed. The sensitivity factor with respect to resistance for member 2 is α_{f_y}=0.4079 and the sensitivity factor with respect to load effect is α_F=-0.9130.

Using the reliability indices and correlation coefficients, it is possible to obtain the reliability index for each pair in the parallel system.

Table 7.14: RI and P_f values of the failure element pairs-member 2 in failure state

Member	β_2	$\beta_{i\|2}$	$P_{f_{pair}}$	β_{pair}
7	3.2157	3.1112	6.5063E-04	3.2157
5	3.2157	3.2789	5.2106E-04	3.2789
1	3.2157	3.2532	5.7053E-04	3.2532

The next step is to calculate the equivalent safety margins for the parallel pairs. These limit state equations are shown in Table 7.15.

The same calculations that were performed to obtain the equivalent safety margins for the damaged state of member 7's failure are used here.

Table 7.15: Equivalent safety margins-member 2 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7	0.4080	-0.9130	$G^e_{2\ \&\ 7} = 0.4080f_y - 0.9130F + 3.2157$
2 & 5	0.4121	-0.9113	$G^e_{2\ \&\ 5} = 0.4121f_y - 0.9113F + 3.2789$
2 & 1	0.4101	-0.9121	$G^e_{2\ \&\ 1} = 0.4101f_y - 0.9121F + 3.2532$

Removing member 10: in order to complete the system reliability evaluation at level two, member 10 has to be removed from the undamaged structure. The processes that have to be performed for this state is the same as the ones performed for the damaged states where member 7 and 2 were removed from the intact structure. Member 10 has a ductile failure in tension; therefore, it has to be replaced by a fictitious set of forces as shown in Figure 7.9.

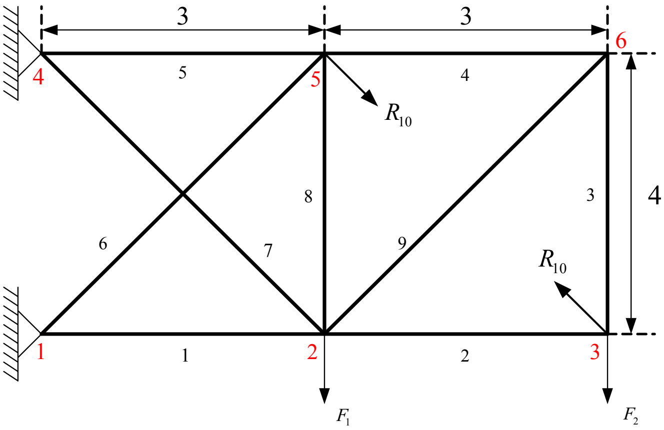

Figure 7.9: 10-bar truss structure with the ductile failure of member 10

The structure shown in Figure 7.9 is analysed elastically to obtain the influence factors for the structure at this state. In the figure, R_{10} is a fictitious load that represents the post-failure resistance of member 10. The influence factors are given in Table 7.16 with respect to the applied loads and R_{10}.

Table 7.16: Influence factors with respect to a_F and $a_{R_{10}}$-member 10 in failure state

Member	a_F	$a_{R_{10}}$
1	-1.5174	0.2238
2	0.0000	-0.4670
3	0.8000	-0.6226
4	0.6000	-0.4670
5	1.0326	0.2238
6	-0.7210	-0.3731
7	1.5290	-0.3731
8	0.5768	-0.3242
9	-1.0000	0.7783

Using the safety factors, the new safety margins for the structure can be formed as shown in table 7.17.

Table 7.17: Safety margins for failure in compression ($G^-_{i|10}$) and tension ($G^+_{i|10}$)-member 10 in failure state

	Safety Margins	
Member	$G^-_{i\|10}$	$G^+_{i\|10}$
1	$G^-_{1\|10} = 1.2296f_y + (-1.5174F + 0.2238f_y)$	$G^+_{1\|10} = 1.9164f_y - (-1.5174F + 0.2238f_y)$
2	$G^-_{2\|10} = 0.4778f_y + (+0.0000F - 0.4670f_y)$	$G^+_{2\|10} = 0.9293f_y - (+0.0000F - 0.4670f_y)$
3	$G^-_{3\|10} = 0.0074f_y + (+0.8000F - 0.6226f_y)$	$G^+_{3\|10} = 0.1885f_y - (+0.8000F - 0.6226f_y)$
4	$G^-_{4\|10} = 0.0131f_y + (+0.6000F - 0.4670f_y)$	$G^+_{4\|10} = 0.1885f_y - (+0.6000F - 0.4670f_y)$
5	$G^-_{5\|10} = 0.6232f_y + (+1.0326F + 0.2238f_y)$	$G^+_{5\|10} = 1.2265f_y - (+1.0326F + 0.2238f_y)$
6	$G^-_{6\|10} = 1.1356f_y + (-0.7210F - 0.3731f_y)$	$G^+_{6\|10} = 2.2704f_y - (-0.7210F - 0.3731f_y)$
7	$G^-_{7\|10} = 0.3115f_y + (+1.5290F - 0.3731f_y)$	$G^+_{7\|10} = 1.0490f_y - (+1.5290F - 0.3731f_y)$
8	$G^-_{8\|10} = 0.0177f_y + (+0.5768F - 0.3242f_y)$	$G^+_{8\|10} = 0.3299f_y - (+0.5768F - 0.3242f_y)$
9	$G^-_{9\|10} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	$G^+_{9\|10} = 0.9293f_y - (-1.0000F + 0.7783f_y)$

The applicable safety margins (failure modes) will be determined and the ones that yield the minimum reliability index will be the governing failure mode as above. The results are presented in Table 7.18.

Table 7.18: Member reliability indices-Member 10 in failure state

Member	Reliability index	Governing Failure Mode
1	3.2324	Compression
2	14.2807	Tension
3	3.5406	Tension
4	3.9567	Tension
5	3.3066	Tension
6	3.7735	Compression
7	3.0796	Tension
8	4.1701	Tension
9	3.4877	Compression
Minimum RI	3.0796	Tension

According to Table 7.18, member 7 has the lowest reliability index, with $\Delta\beta_2$ selected to be 0.2, the interval for the unzipping will be $[3.0796, 3.2796]$. Only members 1 and 7 are the ones with reliability indices within the interval. Hence the parallel pairs will have the form as shown in Figure 7.10.

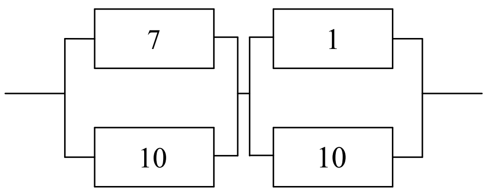

Figure 7.10: Parallel Pairs with member 10

Again, the Parallel pair of members 7 and 10 was already investigated for the damaged state where member 7 was removed from the structure, but it is reached through a different sequence and needs to be investigated. As a result, the the pairs that need to be investigated are the pairs of members 10-7 and 10-1.

Table 7.19: Sensitivity factors and correlation coefficient-member 10 in failure state

Member	α_{f_y}	α_F	$\rho_{i\|10}$
7	0.3991	-0.9169	0.9999
1	0.4091	-0.9125	0.9999

The values in Table 7.19 are calculated using the safety margin of members 1 and 7 in the damaged state where member 10 has failed. The sensitivity factors for member 10 are α_{f_y}=0.4001 and α_F=-0.9165, considering the limit state function of member 10 in the intact structure. From the sensitivity factors the correlation coefficients are calculated as shown in the table above.

Now, the reliability index for the parallel pairs of members 10-1 and 7-10 can be computed. It is shown in Table 7.20.

Table 7.20: RI and P_f values of the failure element pairs-member 10 in failure state

Member	β_{10}	$\beta_{i\|10}$	$P_{f_{pair}}$	β_{pair}
7	3.0960	3.0796	9.7783E-04	3.0969
1	3.0960	3.2324	6.1378E-04	3.2324

The equivalent safety margin for the pair of elements 10 and 1 can be obtained using the data in Table 7.20. The result is presented in Table 7.21.

Table 7.21: Equivalent safety margins-member 10 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
10 & 7	0.3998	-0.9166	$G^e_{10\ \&\ 7} = 0.3998f_y - 0.9166F + 3.0969$
10 & 1	0.4088	-0.9126	$G^e_{10\ \&\ 1} = 0.4088f_y - 0.9126F + 3.2324$

Now, the β-unzipping method at level two is completed and all of the parallel pairs for the system modelling are identified. It is shown in Figure 7.11.

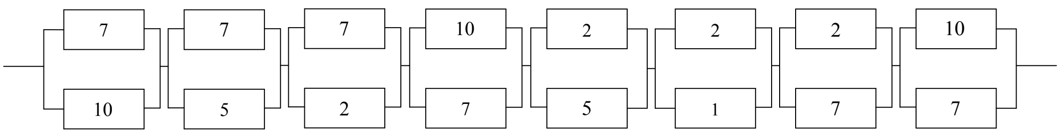

Figure 7.11: Ten-bar truss structure modelled as parallel-series system at level 2

As is seen in Figure 7.11 the whole structure is modelled as a parallel-series system. The model can be used to compute the system reliability of the structure. An outline of the equivalent safety margins for the parallel pairs as well as the reliability indices of the pairs are presented in Table 7.22. Using these reliability indices and safety margins, the reliability of the series system that is composed of parallel pairs can be obtained which is equal to the reliability of the whole structure at level 2.

Table 7.22: Critical parallel pairs at level 2 and their equivalent safety margins and reliability indices

Parallel pairs	Equivalent limit state	Reliability index
$7 \rightarrow 10$	$G^e_{7,10} = 0.4135f_y - 0.9105F + 3.0942$	3.0942
$7 \rightarrow 5$	$G^e_{7,5} = 0.4084f_y - 0.9128F + 3.2003$	3.2003
$7 \rightarrow 2$	$G^e_{7,2} = 0.4077f_y - 0.9131F + 3.1879$	3.1879
$2 \rightarrow 5$	$G^e_{2,5} = 0.4121f_y - 0.9113F + 3.2789$	3.2789
$2 \rightarrow 1$	$G^e_{2,1} = 0.4101f_y - 0.9121F + 3.2532$	3.2532
$2 \rightarrow 7$	$G^e_{2,7} = 0.4080f_y - 0.9130F + 3.2157$	3.2157
$10 \rightarrow 1$	$G^e_{10,1} = 0.4088f_y - 0.9126F + 3.2324$	3.2324
$10 \rightarrow 7$	$G^e_{10,7} = 0.3998f_y - 0.9166F + 3.0969$	3.0969

Based on the sensitivity factors calculated for the parallel pairs shown in Figure 7.11, the correlation matrix can be calculated. It is clear from the sensitivity factors that the equivalent safety margins of the parallel systems are fully correlated. This is confirmed by the result obtained for the correlation matrix.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

Using the reliability indices and the correlation matrix, it is possible to calculate the system reliability of the structure. In order to calculate the system reliability the Ditlevsen bounds for the series system (a series system that is composed of parallel pairs of failure elements) shown in Figure 7.11 can be used. Because of the fact that the equivalent safety margins of the parallel systems are fully correlated the lower Ditlevesen bound will correspond to the system reliability which is equal to the minimum of reliability indices (or maximum of failure probabilities). This is shown below.

$$\beta_s^2 = 3.0942$$

7.2.1.4 System reliability analysis at level three:

System reliability of the truss structure should also be calculated at level three. At this level, the most critical pairs are identified (using the information from system reliability analysis at level two), and the components of the pair are removed from the structure. The removed members are then replaced with a set of fictitious loads representing their post-failure capacity. The damaged structure is analysed elastically, and the influence factors are obtained with respect to the fictitious loads and the external loads applied on the structure. Using the influence factors and resistances of the members, the reliability index for the remaining members can be calculated the same way that was performed at level two. At this stage, the member with the minimum reliability index ($\beta_{\min}$) is identified which is used to determine an interval $[\beta_{\min}, \beta_{\min} + \Delta\beta_3]$ where $\Delta\beta_3$ is chosen arbitrarily. The members with reliability indices within the interval are used to make the so-called triples of failure elements with the two components that were removed from the structure. For each one of the triple parallel systems that are identified through the β-unzipping method the correlation matrix, probability of failure as well as the equivalent safety margin have to be identified which makes the calculation of the system reliability possible.

At level two the following pairs of failure elements were identified. These pairs are shown with their failure modes and reliability indices.

Table 7.23: Pairs of failure elements at level two

Pair of i & j	Failure mode of i	Failure mode of j	Reliability index
7 & 10	Tension	Tension	3.0942
7 & 5	Tension	Tension	3.2003
7 & 2	Tension	Compression	3.1879
2 & 5	Compression	Tension	3.2789
2 & 1	Compression	Compression	3.2532
2 & 7	Compression	Tension	3.2157
10 & 1	Tension	Compression	3.2324
10 & 7	Tension	Tension	3.0969

Removing members 7 and 10: As shown in Table 7.23, the failure pair of members 7 and 10 has the lowest reliability index. Therefore, this pair will be used as the first pair for the system reliability evaluation at level 3. This means that members 7 and 10 should be removed from the structure. Enough attention should be drawn towards the failure modes of the members. In this case both of the elements are failing in tension. This damaged state of the structure is shown in Figure 7.12.

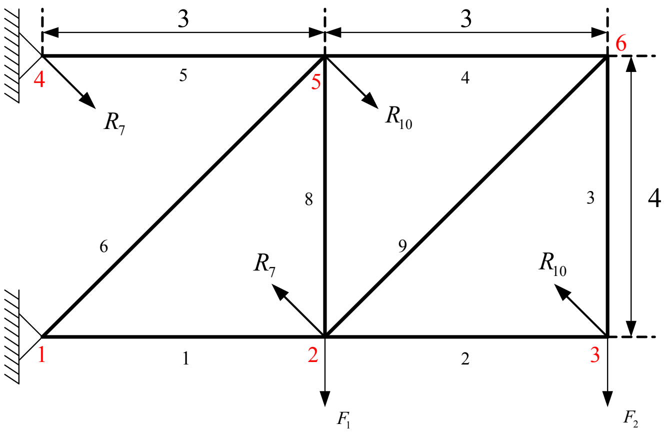

Figure 7.12: Damaged state of the structure with the ductile failure of elements 7 & 10

The structure shown in Figure 7.12 should be analysed elastically with respect to the fictitious loads (R_7 and R_{10}) as well as the applied loads ($F_1 = F$ and $F_2 = 0.8F$) in order to calculate the influence coefficients for the members of the structure. As was done for level 2, each time a load of unit value is applied on the structure for each of the loads and through the elastic analysis the influence factors are obtained. These values are presented in Table 7.24 below.

Table 7.24: Influence factors with respect to F, R_7^+, and R_{10}^+-members 7 & 10 in failure state

	Influence factors		
Member	a_F	a_{R_7}	$a_{R_{10}}$
1	-0.6000	-0.6294	0.0000
2	0.0000	0.0000	-0.4670
3	0.8000	0.0000	-0.6226
4	0.6000	0.0000	-0.4670
5	1.9500	-0.6294	0.0000
6	-2.2500	1.0490	0.0000
8	1.8000	-0.8392	-0.6226
9	-1.0000	0.0000	0.7783

Using the influence factors, it is possible to form the limit states for the remaining components of the structure. These are shown in Table 7.25.

Table 7.25: Safety margins for failure in compression ($G^-_{i|10,7}$) and tension ($G^+_{i|10,7}$)-members 7 & 10 in failure state

	Safety Margins	
Member	$G^-_{i\|10,7}$	$G^+_{i\|10,7}$
1	$G^-_{1\|10,7} = 1.2296f_y + (-0.6000F - 0.6294f_y)$	$G^+_{1\|10,7} = 1.9164f_y - (-0.6000F - 0.6294f_y)$
2	$G^-_{2\|10,7} = 0.4778f_y + (+0.0000F - 0.4670f_y)$	$G^+_{2\|10,7} = 0.9293f_y - (+0.0000F - 0.4670f_y)$
3	$G^-_{3\|10,7} = 0.0074f_y + (+0.8000F - 0.6226f_y)$	$G^+_{3\|10,7} = 0.1885f_y - (+0.8000F - 0.6226f_y)$
4	$G^-_{4\|10,7} = 0.0131f_y + (+0.6000F - 0.4670f_y)$	$G^+_{4\|10,7} = 0.1885f_y - (+0.6000F - 0.4670f_y)$
5	$G^-_{5\|10,7} = 0.6232f_y + (+1.9500F - 0.6294f_y)$	$G^+_{5\|10,7} = 1.2265f_y - (+1.9500F - 0.6294f_y)$
6	$G^-_{6\|10,7} = 1.1356f_y + (-2.2500F + 1.0490f_y)$	$G^+_{6\|10,7} = 2.2704f_y - (-2.2500F + 1.0490f_y)$
8	$G^-_{8\|10,7} = 0.0177f_y + (+1.8000F - 1.4618f_y)$	$G^+_{8\|10,7} = 0.3299f_y - (+1.8000F + 1.4618f_y)$
9	$G^-_{9\|10,7} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	$G^+_{9\|10,7} = 0.9293f_y - (-1.0000F + 0.7783f_y)$

In the limit state equations, since both of the resistances are in terms of the yield capacity of the failed members, they were summed. However, attention is drawn to Table 7.24 which shows that for most of the members the influence factors are zero with respect to one of the resistances.

The safety margins are used to calculate the reliability indices for the members. The results are shown in Table 7.26.

Table 7.26: Member Reliability indices-members 7 & 10 failed

Member	Reliability index	Governing Failure Mode
1	3.4673	Compression
2	14.2807	Tension
3	3.5406	Compression
4	3.9567	Compression
5	3.2003	Compression
6	3.3063	Tension
8	3.4402	Compression
9	3.4877	Tension
Minimum RI	3.2003	Compression

For the determination of the governing failure mode the applicability of a certain failure mode has to be verified as was done in level 2. It is seen in Table 7.26 that member 5 has the lowest reliability index. If $\Delta\beta_3$ is selected to be 0.15, the interval that can be used for the unzipping at this stage is $[3.2003, 3.3503]$.

From the interval it can be seen that only the reliability indices of members 5 and 6 fall within the interval which will lead to the following triples of parallel elements as shown in Figure 7.13.

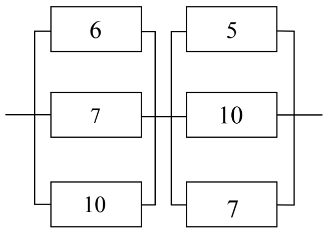

Figure 7.13: Triples of failure elements (element 7 & 10 failed)

The sensitivity values for members 6 and 5 are determined for the damaged structure (members 7 & 10 failed). Also the correlation coefficients with respect to members 7 and 10 are given.

Table 7.27: Sensitivity factors and correlation coefficient-members 7 & 10 in failure state

Member	α_{f_y}	α_F	$\rho_{i,7}$	$\rho_{i,10}$
5	0.4069	-0.9135	0.9996	0.9999
6	0.4137	-0.9104	0.9999	0.9998

The correlation coefficient between the safety margins of members 7 and 10 was calculated in system reliability evaluation at level two which is equal to 0.9999.

Table 7.28: RI and P_f values for the triples of failure elements-members 7 & 10 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
7 & 10 & 5	3.0797	3.0928	3.2003	6.8642E-04	3.2003
7 & 10 & 6	3.0797	3.0928	3.3063	4.7268E-04	3.3063

Now it is possible to calculate the equivalent safety margins for each one of the triples. The results of the calculations are shown in Table 7.29.

Table 7.29: Equivalent safety margins-members 7 & 10 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
7 & 10 & 5	0.4072	-0.9134	$G^e_{7\ \&\ 10\ \&\ 5} = 0.4072 f_y - 0.9134 F + 3.2003$
7 & 10 & 6	0.4141	-0.9102	$G^e_{7\ \&\ 10\ \&\ 6} = 0.4141 f_y - 0.9102 F + 3.3063$

It was shown in level 2 that the damaged state where members 7 and 10 have failed also refers to the sequence of elements 10 and 7. Therefore, the same calculations need to be performed for the case of the sequence of members 10 and 7.

Table 7.30 below shows the correlation coefficients as well the sensitivity factors.

Table 7.30: Sensitivity factors and correlation coefficient-members 10 & 7 in failure state

Member	α_{f_y}	α_F	$\rho_{i,10}$	$\rho_{i,7}$
5	0.4069	-0.9135	0.9999	0.9999
6	0.4137	-0.9104	0.9999	0.9999

The correlation coefficient between the safety margins of members 10 and 7 was calculated to be 0.9999. In Table 7.31 the reliability indices of the parallel triples are given.

Table 7.31: RI and P_f values for the triples of failure elements-members 10 & 7 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
10 & 7 & 5	3.0960	3.0796	3.2003	6.8642E-04	3.2003
10 & 7 & 6	3.0797	3.0796	3.3063	4.7268E-04	3.3063

the equivalent safety margins are shown in Table 7.32 below.

Table 7.32: Equivalent safety margins-members 10 & 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
10 & 7 & 5	0.4072	-0.9134	$G^e_{10\ \&\ 7\ \&\ 5} = 0.4072 f_y - 0.9134F + 3.2003$
10 & 7 & 6	0.4141	-0.9102	$G^e_{10\ \&\ 7\ \&\ 6} = 0.4141 f_y - 0.9102F + 3.3063$

It can be seen that due the high correlation between the safety margins of elements and the fact that for both of the sequences the reliability index values of elements 10 and 7 were close, the result of the sequences of elements 7-10 and 10-7 led to the same equivalent sensitivity factors and reliability indices. Nevertheless, that might not always be the case and the different sequence need to considered separately.

Removing members 7 and 2: The next critical pair is the pair of elements 7 and 2. These members are removed from the structure. This is shown in Figure 7.14 below. It should also be mentioned that for this pair there are also two sequences that lead to this damaged state. One is the sequence of 7 and 2 and the other is the sequence of 2 and 7.

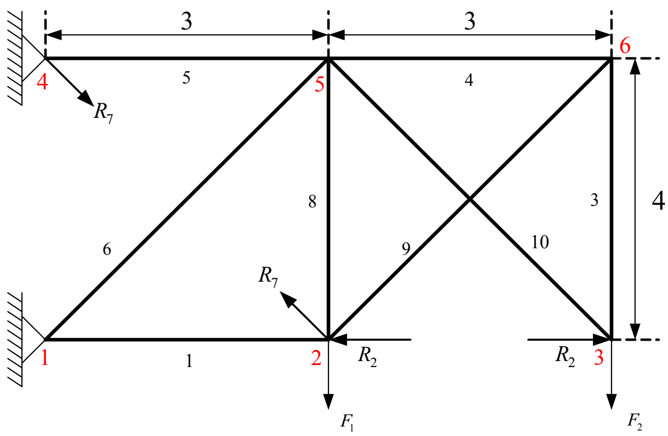

Figure 7.14: Damaged state of the structure with the ductile failure of elements 7 & 2

As shown in Figure 7.14 member 7 fails in tension and member 2 in compression. The influence coefficients for the damaged state of the structure are shown in Table 7.33.

Table 7.33: Influence factors with respect to F, R_2^-, and R_7^+-members 7 & 2 in failure state

	Influence factors		
Member	a_F	a_{R_7}	a_{R_2}
1	-0.6000	0.0000	-0.6294
3	0.8000	-0.6371	0.0000
4	0.6000	-0.4778	0.0000
5	1.9500	0.0000	-0.6294
6	-2.2500	0.0000	1.0490
8	1.8000	-0.6371	-0.8392
9	-1.0000	0.7963	0.0000
10	0.0000	0.7963	0.0000

With the influence factors known, the safety margins for the remaining members of the structure can be obtained. The safety margins are presented in Table 7.34.

Table 7.34: Safety margins for failure in compression ($G^-_{i|2,7}$) and tension ($G^+_{i|2,7}$)-members 7 & 2 in failure state

	Safety Margins	
Member	$G^-_{i\|2,7}$	$G^+_{i\|2,7}$
1	$G^-_{1\|7,2} = 1.2296f_y + (-0.6000F - 0.6294f_y)$	$G^+_{1\|7,2} = 1.9164f_y - (-0.6000F - 0.6294f_y)$
3	$G^-_{3\|7,2} = 0.0074f_y + (+0.8000F - 0.6371f_y)$	$G^+_{3\|7,2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$
4	$G^-_{4\|7,2} = 0.0131f_y + (+0.6000F - 0.4778f_y)$	$G^+_{4\|7,2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$
5	$G^-_{5\|7,2} = 0.6232f_y + (+1.9500F - 0.6294f_y)$	$G^+_{5\|7,2} = 1.2265f_y - (+1.9500F - 0.6294f_y)$
6	$G^-_{6\|7,2} = 1.1356f_y + (-2.2500F + 1.0490f_y)$	$G^+_{6\|7,2} = 2.2704f_y - (-2.2500F + 1.0490f_y)$
8	$G^-_{8\|7,2} = 0.0177f_y + (+1.8000F - 1.4763f_y)$	$G^+_{8\|7,2} = 0.3299f_y - (+1.8000F + 1.4763f_y)$
9	$G^-_{9\|7,2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	$G^+_{9\|7,2} = 0.9293f_y - (-1.0000F + 0.7783f_y)$
10	$G^-_{10\|7,2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	$G^+_{10\|7,2} = 0.7783f_y - (+0.0000F + 0.7963f_y)$

The following reliability indices are calculated for the remaining members.

Table 7.35: Member reliability indices-Members 7 and 2 in failure state

Member	Reliability index	Governing Failure Mode
1	3.4673	Compression
3	3.6375	Tension
4	4.0501	Tension
5	3.2003	Tension
6	3.3063	Compression
8	3.4838	Tension
9	3.5850	Compression
10	14.2807	Compression
Minimum RI	3.2003	Compression

From the results in Table 7.35, it is clear that member 5 has the lowest reliability index which will lead to the interval $[3.2003, 3.3503]$ with the assumption of $\Delta\beta_3$ equal 0.15. Only members 5 and 6 have reliabilities within the interval. Parallel triples of failure elements are shown.

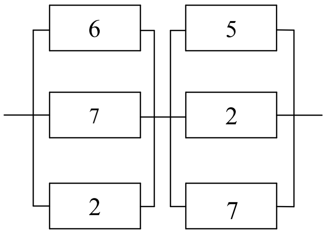

Figure 7.15: Triples of failure elements (elements 7 & 2 failed)

The sensitivity factors of member 6 and 5 in the damaged state, where members 2 and 7 have failed, are given in Table 7.36 below.

Table 7.36: Sensitivity factors and correlation coefficient-members 7 & 2 in failure state

Member	α_{f_y}	α_F	$\rho_{i,7}$	$\rho_{i,2}$
5	0.4069	-0.9135	0.9999	0.9999
6	0.4137	-0.9104	0.9999	0.9999

The correlation coefficient between the safety margins of members 2 and 7 is 0.9990. The reliability indices for the parallel triples of failure elements are given in Table 7.37.

Table 7.37: RI and P_f for the triples of failure elements-members 7 & 2 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
7 & 2 & 5	3.0797	3.1879	3.2003	6.8282E-04	3.2018
7 & 2 & 6	3.0797	3.1879	3.3063	4.7268E-04	3.3063

Using the results presented in Tables 7.36 and Table 7.37, the equivalent safety margins are calculated as shown in Table 7.38.

Table 7.38: Equivalent safety margins-members 7 & 2 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
7 & 2 & 5	0.4072	-0.9134	$G^e_{7\ \&\ 2\ \&\ 5} = 0.4072 f_y - 0.9134F + 3.2018$
7 & 2 & 6	0.4121	-0.9102	$G^e_{7\ \&\ 2\ \&\ 6} = 0.4121 f_y - 0.9102F + 3.3063$

The sequence of elements 2-7-5 as well as the sequence of elements 2-7-6 also needs to be investigated. Table 7.39 shows the sensitivity factors and correlation coefficients.

Table 7.39: Sensitivity factors and correlation coefficient-members 2 & 7 in failure state

Member	α_{f_y}	α_F	$\rho_{i,2}$	$\rho_{i,7}$
5	0.4069	-0.9135	0.9999	0.9999
6	0.4137	-0.9104	0.9999	0.9999

In Table 7.40 the reliability index value of the triples are shown as well as the probabilities of failure.

Table 7.40: RI and P_f for the triples of failure elements-members 2 & 7 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
2 & 7 & 5	3.2175	3.1112	3.2003	6.4479E-04	3.2183
2 & 7 & 6	3.2175	3.1112	3.3063	4.7268E-04	3.3063

Table 7.41 the equivalent safety margins for the parallel triples of elements 2-7-5 and 2-7-6 are depicted.

Table 7.41: Equivalent safety margins-members 2 & 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7 & 5	0.4080	-0.9130	$G^e_{2\ \&\ 7\ \&\ 5} = 0.4080f_y - 0.9130F + 3.2183$
2 & 7 & 6	0.4141	-0.9102	$G^e_{2\ \&\ 7\ \&\ 6} = 0.4141f_y - 0.9102F + 3.3063$

Removing members 5 and 7: The third pair of elements that have to be removed from the structure is the pair of elements 5 & 7. However, it is evident that the damaged state where both of the members 7 and 5 have failed will cause the failure of the whole structure (a mechanism will be formed). Therefore, this damaged state is not applicable to the system reliability analysis at level 3.

Removing members 10 and 1: The next pair of failure elements that can be removed from the structure is the pair of elements 1 and 10. This damaged state is shown in Figure 7.16.

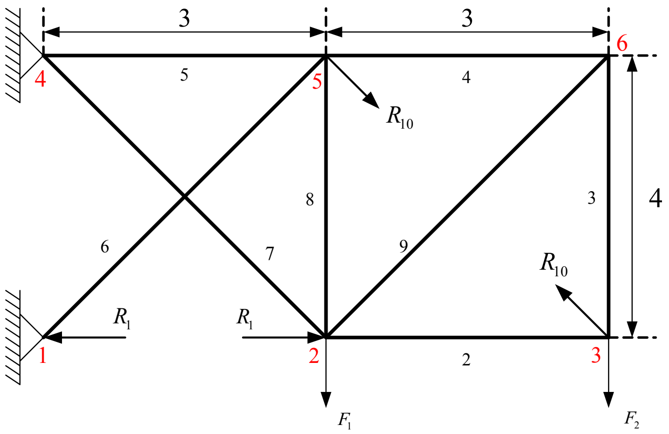

Figure 7.16: Damaged state of the structure with the ductile failure of elements 10 & 1

As shown in the Figure, element 1 fails in compression and element 10 fails in tension. The influence factors for this damaged state of the structure are shown in Table 7.42.

Table 7.42: Influence factors with respect to F, R_1^-, and R_{10}^+-members 10 & 1 in failure state

	Influence factors		
Member	a_F	a_{R_1}	$a_{R_{10}}$
2	0.0000	0.0000	-0.4670
3	0.8000	0.0000	-0.6226
4	0.6000	0.0000	-0.4670
5	2.5500	-1.2296	0.0000
6	-3.2500	2.0493	0.0000
7	-1.0000	2.0493	0.0000
8	2.6000	-1.6395	-0.6226
9	-1.0000	0.0000	0.7783

The limit state functions of the members for the damaged state of the structure are as shown in Table 7.43.

Table 7.43: Safety margins for failure in compression ($G^-_{i|1,10}$) and tension ($G^+_{i|1,10}$)-members 10 & 1 in failure state

	Safety Margins	
Member	$G^-_{i\|1,10}$	$G^+_{i\|1,10}$
2	$G^-_{2\|1,10} = 0.4778f_y + (+0.0000F - 0.4670f_y)$	$G^+_{2\|1,10} = 0.9293f_y - (+0.0000F - 0.4670f_y)$
3	$G^-_{3\|1,10} = 0.0074f_y + (+0.8000F - 0.6226f_y)$	$G^+_{3\|1,10} = 0.1885f_y - (+0.8000F - 0.6226f_y)$
4	$G^-_{4\|1,10} = 0.0131f_y + (+0.6000F - 0.4670f_y)$	$G^+_{4\|1,10} = 0.1885f_y - (+0.6000F - 0.4670f_y)$
5	$G^-_{5\|1,10} = 0.6232f_y + (+2.5500F - 1.2296f_y)$	$G^+_{5\|1,10} = 1.2265f_y - (+2.5500F - 1.2296f_y)$
6	$G^-_{6\|1,10} = 1.1356f_y + (-3.2500F + 2.0493f_y)$	$G^+_{6\|1,10} = 2.2704f_y - (-3.2500F + 1.0493f_y)$
7	$G^-_{7\|1,10} = 0.3115f_y + (-1.0000F + 2.0493f_y)$	$G^+_{7\|1,10} = 1.0490f_y - (-1.0000F + 2.0493f_y)$
8	$G^-_{8\|1,10} = 0.0177f_y + (+2.6000F - 2.2621f_y)$	$G^+_{8\|1,10} = 0.3299f_y - (+2.6000F - 2.2621f_y)$
9	$G^-_{9\|1,10} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	$G^+_{9\|1,10} = 0.9293f_y - (-1.0000F + 0.7783f_y)$

The reliability indices for the damaged state of the structure are given in Table 7.44.

Table 7.44: Member reliability indices-members 10 & 1 in failure state

Member	Reliability index	Governing Failure Mode
2	14.2807	Tension
3	3.5406	Tension
4	3.9567	Tension
5	3.2636	Tension
6	3.3101	Compression
7	8.6704	Compression
8	3.4485	Tension
9	3.4877	Compression
Minimum RI	3.2636	Tension

From the calculated reliability indices in Table 7.44, the minimum reliability index equals 3.2636. With the assumption of $\Delta\beta_3$=0.15, the unzipping interval of $[3.2636, 3.4136]$ will be obtained. This means the parallel triple of elements 1-10-5 and 1-10-6 should be formed as shown in Figure 7.17.

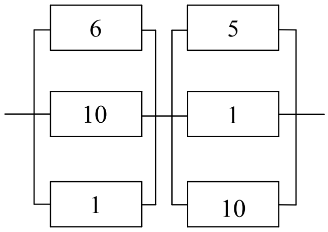

Figure 7.17: Triples of failure elements (elements 1 & 10 failed)

The sensitivity factors and correlation coefficient are shown in Table 7.45 below.

Table 7.45: Sensitivity factors and correlation coefficient-members 10 & 1 in failure state

Member	α_{f_y}	α_F	$\rho_{i,10}$	$\rho_{i,1}$
5	0.4109	-0.9117	0.9999	0.9999
6	0.4169	-0.9090	0.9999	0.9999

The sensitivity factors for members 1 and 10 are $\bar{\alpha}_{10} = [0.4001, -0.9165]$ and $\bar{\alpha}_{1|10} = [0.4090, -0.9125]$. The correlation between the safety margins of member 1 and 10 is also 0.9999.

The reliability indices and probabilities of failure for the parallel systems are presented in Table 7.46.

Table 7.46: RI and P_f values for the triples of failure elements-members 10 & 1 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
10 & 1 & 5	3.0960	3.2324	3.2636	5.4989E-04	3.2637
10 & 1 & 6	3.0960	3.2324	3.3560	3.9539E-04	3.3560

The equivalent safety margins are computed for the two parallel systems in Figure 7.17. These safety margins are shown in Table 7.47.

Table 7.47: Equivalent safety margins-members 10 & 1 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
1 & 10 & 5	0.4109	-0.9117	$G^e_{1\ \&\ 10\ \&\ 5} = 0.4109f_y - 0.9117F + 3.2637$
1 & 10 & 6	0.4170	-0.9089	$G^e_{1\ \&\ 10\ \&\ 6} = 0.4170f_y - 0.9089F + 3.3560$

Removing members 2 and 1: Next pair of elements that have to be removed from the structure is the pair of elements 1 and 2. Both of the elements are failing in compression. The damaged state of the structure is shown in the Figure 7.18 below.

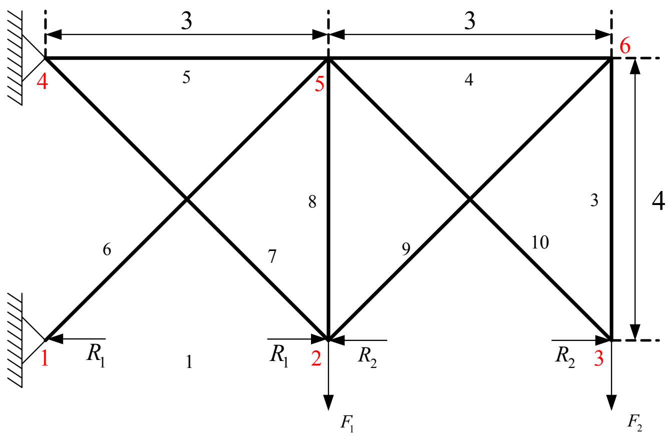

Figure 7.18: Damaged state of the structure with elements 1 and 2 failed

For this damaged state of the structure the following influence factors are computed.

Table 7.48: Influence factors with respect to F, R_1^-, and R_2^--members 2 & 1 in failure state

	Influence factors		
Member	a_F	a_{R_1}	a_{R_2}
3	0.8000	0.0000	-0.6371
4	0.6000	0.0000	-0.4778
5	2.5500	-1.2296	0.0000
6	-3.2500	2.0493	0.0000
7	-1.0000	2.0493	0.0000
8	2.6000	-1.6395	-0.6371
9	-1.0000	0.0000	0.7963
10	0.0000	0.0000	0.7963

Using the influence factors shown in Table 7.48, the safety margins are calculated as shown in Table 7.49.

Table 7.49: Safety margins for failure in compression ($G^-_{i|1,2}$) and tension ($G^+_{i|1,2}$)-members 2 & 1 in failure state

	Safety Margins	
Member	$G^-_{i\|1,2}$	$G^+_{i\|1,2}$
3	$G^-_{3\|1,2} = 0.0074f_y + (+0.8000F - 0.6371f_y)$	$G^+_{3\|1,2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$
4	$G^-_{4\|1,2} = 0.0131f_y + (+0.6000F - 0.4778f_y)$	$G^+_{4\|1,2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$
5	$G^-_{5\|1,2} = 0.6232f_y + (+2.5500F - 1.2296f_y)$	$G^+_{5\|1,2} = 1.2265f_y - (+2.5500F - 1.2296f_y)$
6	$G^-_{6\|1,2} = 1.1356f_y + (-3.2500F + 2.0493f_y)$	$G^+_{6\|1,2} = 2.2704f_y - (-3.2500F + 1.0493f_y)$
7	$G^-_{7\|1,2} = 0.3115f_y + (-1.0000F + 2.0493f_y)$	$G^+_{7\|1,2} = 1.0490f_y - (-1.0000F + 2.0493f_y)$
8	$G^-_{8\|1,2} = 0.0177f_y + (+2.6000F - 2.2766f_y)$	$G^+_{8\|1,2} = 0.3299f_y - (+2.6000F - 2.2766f_y)$
9	$G^-_{9\|1,2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	$G^+_{9\|1,2} = 0.9293f_y - (-1.0000F + 0.7963f_y)$
10	$G^-_{10\|1,2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	$G^+_{10\|1,2} = 0.7783f_y - (+0.0000F + 0.7963f_y)$

The reliability indices for the damaged state of the structure are calculated. These values are shown in Table 7.50.

Table 7.50: Member reliability indices-members 2 and 1 in failure state

Member	Reliability index	Governing Failure Mode
3	3.6375	Tension
4	4.0501	Tension
5	3.2636	Tension
6	3.3560	Compression
7	8.6704	Compression
8	3.4786	Tension
9	3.5850	Compression
10	14.2807	Compression
Minimum RI	3.2636	Tension

From the results shown in Table 7.50, it is clear that the minimum reliability index belongs to member 5. Therefore, the interval will be $[3.2636, 3.4136]$.

Only members 5 and 6 have the reliability values within the interval. The parallel triples of failure elements are shown in Figure 7.19.

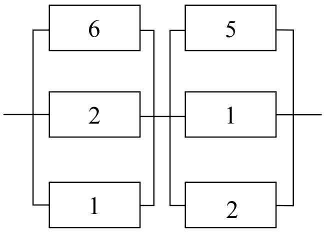

Figure 7.19: Triples of failure elements (elements 2 & 1 failed)

Sensitivity factors and correlation coefficients for the parallel triples of failure elements are shown in Table 7.51.

Table 7.51: Sensitivity factors and correlation coefficient-members 2 & 1 in failure state

Member	α_{f_y}	α_F	$\rho_{i,2}$	$\rho_{i,1}$
5	0.4109	-0.9117	0.9999	0.9999
6	0.4169	-0.9090	0.9999	0.9999

The sensitivity factors for elements 2 and 1 are $\bar{\alpha}_2 = [0.4079, 0.9130]$ and $\bar{\alpha}_{1|2} = [0.4102, -0.9120]$, and the correlation between the safety margins of elements 1 and 2 is 0.9999.

The reliability indices of the parallel systems are given in Table 7.52.

Table 7.52: RI and P_f values for the triples of failure elements-members 2 & 1 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{triple}}$	β_{triple}
2 & 1 & 5	3.2157	3.2532	3.2636	5.4628E-04	3.2655
2 & 1 & 6	3.2157	3.2532	3.3560	3.9539E-04	3.3560

Based on the results presented in Table 7.52, the equivalent safety margins are calculated for the parallel triples of Figure 7.19.

Table 7.53: Equivalent safety margins-members 2 & 1 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
1 & 2 & 5	0.4112	-0.9115	$G^e_{1\ \&\ 2\ \&\ 5} = 0.4112f_y - 0.9115F + 3.2655$
1 & 2 & 6	0.4170	-0.9089	$G^e_{1\ \&\ 2\ \&\ 6} = 0.4170f_y - 0.9089F + 3.3560$

Removing members 2 and 5: The only other pair of failure elements that need to be removed from the structure is the pair of elements 2 and 5. Element 2 fails in compression and element 5 fails in tension. These two members are replaced by a set forces representing their post failure capacity. This damaged state of the structure is shown in Figure 7.20 below.

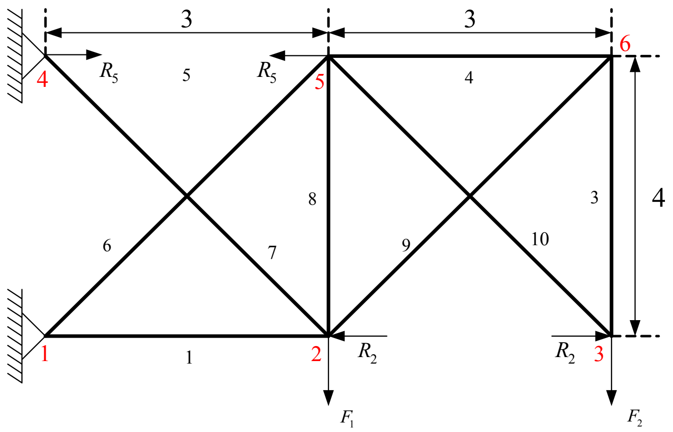

Figure 7.20: Damaged state of the structure with elements 2 & 5 failed

For the damaged state where members 2 and 5 have failed the following influence factors are calculated with respect to the applied loads and the fictitious loads:

Table 7.54: Influence factors with respect to F, R_2^-, and R_5^+-members 2 & 5 in failure state

	Influence factors		
Member	a_F	a_{R_2}	a_{R_5}
1	-2.5500	0.0000	1.2265
3	0.8000	-0.6371	0.0000
4	0.6000	-0.4778	0.0000
6	1.0000	0.0000	-2.0442
7	3.2500	0.0000	-2.0442
8	-0.8000	-0.6371	1.6353
9	-1.0000	0.7963	0.0000
10	0.0000	0.7963	0.0000

The safety margins for the damaged state of the structure are shown in Table 7.55.

Table 7.55: Safety margins for failure in compression ($G^-_{i|2,5}$) and tension ($G^+_{i|2,5}$)-members 2 & 5 in failure state

	Safety Margins	
Member	$G^-_{i\|2,5}$	$G^+_{i\|2,5}$
1	$G^-_{1\|2,5} = 1.2296f_y + (-2.5500F + 1.2265f_y)$	$G^+_{1\|2,5} = 1.9164f_y - (-2.5500F + 1.2265f_y)$
3	$G^-_{3\|2,5} = 0.0074f_y + (+0.8000F - 0.6371f_y)$	$G^+_{3\|2,5} = 0.1885f_y - (+0.8000F - 0.6371f_y)$
4	$G^-_{4\|2,5} = 0.0131f_y + (+0.6000F - 0.4778f_y)$	$G^+_{4\|2,5} = 0.1885f_y - (+0.6000F - 0.4778f_y)$
6	$G^-_{6\|2,5} = 1.1356f_y + (+1.0000F - 2.0444f_y)$	$G^+_{6\|2,5} = 2.2704f_y - (+1.0000F - 2.0442f_y)$
7	$G^-_{7\|2,5} = 0.3115f_y + (+3.2500F - 2.0442f_y)$	$G^+_{7\|2,5} = 1.0490f_y - (+3.2500F - 2.0442f_y)$
8	$G^-_{8\|2,5} = 0.0177f_y + (-0.8000F + 0.9982f_y)$	$G^+_{8\|2,5} = 0.3299f_y - (-0.8000F + 0.9982f_y)$
9	$G^-_{9\|2,5} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	$G^+_{9\|2,5} = 0.9293f_y - (-1.0000F + 0.7963f_y)$
10	$G^-_{10\|2,5} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	$G^+_{10\|2,5} = 0.7783f_y - (+0.0000F + 0.7963f_y)$

The reliability indices are calculated using the relevant safety margins. These values are demonstrated in Table 7.56.

Table 7.56: Member reliability indices-members 2 & 5 in failure state

Member	Reliability index	Governing Failure Mode
1	3.2637	Compression
3	3.6375	Tension
4	4.0501	Tension
6	11.5292	Tension
7	3.2004	Tension
8	4.8397	Compression
9	3.5850	Compression
10	14.2807	Compression
Minimum RI	3.2004	Tension

From the reliability values shown in Table 7.56, it can be seen that member 7 has the smallest reliability index value which is equal to 3.2004. With the assumption of $\Delta\beta_3$ equal to 0.15, the interval for this damaged state of the structure is $[3.2004, 3.3504]$.

The only components with the reliability indices within the interval are members 1 and 7. This will lead to the following parallel systems which are shown in Figure 7.21.

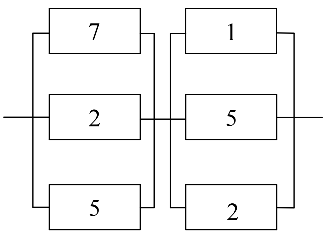

Figure 7.21: Triples of failure elements (elements 2 & 5 failed)

The parallel systems shown in Figure 7.21 are repetitive. Nevertheless, the are reached through different sequences and need to be taken into account.

Table 7.57 below shows the sensitivity factors for the parallel triples of elements.

Table 7.57: Sensitivity factors and correlation coefficient-members 2 & 5 in failure state

Member	α_{f_y}	α_F	$\rho_{i,2}$	$\rho_{i,1}$
5	0.0.4121	-0.9112	0.9999	0.9999
6	0.0.4121	-0.9112	0.9999	0.9999

The sensitivity factor vector for element 2 is $\bar{\alpha}_2 = [0.4079, 0.9130]$ and the sensitivity factor vector for element 5 $\bar{\alpha}_{5|2} = [0.4119, 0.9112]$. Also the correlation coefficient between the safety margins of elements 2 and 5 is 0.9999.

The reliability index values for the elements and the parallel triples of failure element are given below.

Table 7.58: RI and P_f values for the triples of failure elements-members 2 & 5 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
2 & 5 & 7	3.2157	3.2789	3.2500	5.2088E-04	3.2790
2 & 5 & 1	2.2157	3.2789	3.2637	5.1913E-04	3.2799

Using the data provided in Tables 7.57 and 7.58, it is possible to get the equivalent safety margins. These are shown in Table 7.59 below.

Table 7.59: Equivalent safety margins-members 2 and 5 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 5 & 7	0.4112	-0.9112	$G^e_{2\ \&\ 5\ \&\ 7} = 0.4112 f_y - 0.9112 F + 3.2790$
2 & 5 & 1	0.4112	-0.9112	$G^e_{2\ \&\ 5\ \&\ 1} = 0.4112 f_y - 0.9112 F + 3.2799$

All of the possible parallel systems for the structure were identified for the system reliability analysis at level 3. Consequently, the whole structure can be modelled as a series system whose elements are triples of parallel elements. In other words, the structure is modelled as parallel-series system. This system reliability model for the structure is shown in Figure 7.22.

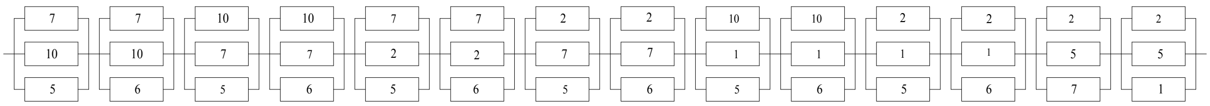

Figure 7.22: Ten-bar truss structure modelled as a parallel-series system at level 3

In Figure 7.22 the parallel-series model that is used for the modelling of the structure is shown. The figure shows 14 parallel systems for this level. Table 7.60 shows the equivalent safety margins, reliability indices, and failure probabilities for the aforementioned parallel systems.

Table 7.60: Critical Parallel triples at level 3 and their equivalent safety margins and reliability indices

Parallel triple	Equivalent limit state	Failure probability	RI
$7 \to 10 \to 5$	$G^e_{7\ \&\ 10\ \&\ 5} = 0.4072f_y - 0.9134F + 3.2003$	6.8642E-04	3.2003
$7 \to 10 \to 6$	$G^e_{7\ \&\ 10\ \&\ 6} = 0.4141f_y - 0.9102F + 3.3063$	4.7268E-04	3.3063
$10 \to 7 \to 5$	$G^e_{7\ \&\ 10\ \&\ 5} = 0.4072f_y - 0.9134F + 3.2003$	6.8642E-04	3.2003
$10 \to 7 \to 6$	$G^e_{7\ \&\ 10\ \&\ 6} = 0.4141f_y - 0.9102F + 3.3063$	4.7268E-04	3.3063
$7 \to 2 \to 5$	$G^e_{7\ \&\ 2\ \&\ 5} = 0.4072f_y - 0.9134F + 3.2018$	6.8282E-04	3.2018
$7 \to 2 \to 6$	$G^e_{7\ \&\ 2\ \&\ 6} = 0.4121f_y - 0.9102F + 3.3063$	4.7268E-04	3.3063
$2 \to 7 \to 5$	$G^e_{2\ \&\ 7\ \&\ 5} = 0.4080f_y - 0.9130F + 3.2183$	6.4479E-04	3.2183
$2 \to 7 \to 6$	$G^e_{2\ \&\ 7\ \&\ 6} = 0.4141f_y - 0.9102F + 3.3063$	4.7268E-04	3.3063
$10 \to 1 \to 5$	$G^e_{1\ \&\ 10\ \&\ 5} = 0.4109f_y - 0.9117F + 3.2637$	5.4989E-04	3.2637
$10 \to 1 \to 6$	$G^e_{1\ \&\ 10\ \&\ 6} = 0.4170f_y - 0.9089F + 3.3560$	3.9539E-04	3.3560
$2 \to 1 \to 5$	$G^e_{1\ \&\ 2\ \&\ 5} = 0.4112f_y - 0.9115F + 3.2655$	5.4628E-04	3.2655
$2 \to 1 \to 6$	$G^e_{1\ \&\ 2\ \&\ 6} = 0.4170f_y - 0.9089F + 3.3560$	3.9539E-04	3.3560
$2 \to 5 \to 7$	$G^e_{2\ \&\ 5\ \&\ 7} = 0.4112f_y - 0.9112F + 3.2790$	5.2088E-04	3.2790
$2 \to 5 \to 1$	$G^e_{2\ \&\ 5\ \&\ 1} = 0.4112f_y - 0.9112F + 3.2799$	5.1913E-04	3.2799

If the Ditlevsen bounds are applied for the series system at this level, the following interval will be obtained for the system reliability of the structure.

The correlation coefficient matrix for this structure is a 14×14 matrix. This is due to the fact that the series model is composed of 14 parallel triples of elements.

$$6.8642 \times 10^{-4} \leq P_f^3 \leq 7.2094 \times 10^{-3}$$

The following correlation coefficient matrix is calculated for the elements of the series system (elements of the series system are parallel triples) using the equivalent sensitivity factors calculated for each of the parallel systems.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

According to correlation coefficient matrix the equivalent safety margins of the elements of the series system are highly correlated. Therefore, the lower Ditlevsen bound is a good estimate for the system reliability. This will yield the following system reliability.

$$\beta_s^3 = 3.2003$$

The structure has two degrees of redundancy (2 degrees of indeterminacy). According to the theory of plasticity, for a structure the number plastic hinges (element failures in trusses) that are required to cause the collapse of the structure equals degree-of-indeterminacy plus one [50]. As a result, for this structure failure of 3 members will cause the collapse of the whole structure. This means it is not possible to go further than level three for this structure. However, in general, for the structures with higher levels of redundancy, reliability investigation can be performed for higher levels [45].

It should also be noted that for this structure the system reliability analysis at level 3 provides the most accurate estimation for the reliability of the whole structure owing to the degree of redundancy of the structure which is equal to two.

In table 7.61 the outline of the system reliability is given for different levels.

Table 7.61: System reliability of the structure for different levels

Level	Reliability index (RI)	Failure Probability
0	3.0797	1.0360E-03
1	3.0797	1.0360E-03
2	3.0942	9.8672E-04
3	3.2003	6.8642E-04

7.2.2 System reliability analysis for non-normal random variables

For the case of non-normal random variables, the same procedure can be followed. The only difference is that the reliability evaluation of the safety margins has to be performed through an iterative method such as the FORM method. As a result, most of the data obtained in section 7.2.1.1 to 7.2.1.4 can be used for this case, such as the influence factors, governing safety margins and failure modes. Therefore, only a reassessment of the safety margins is necessary to obtain the system reliability for this case. The detailed calculations for this case are demonstrated in Appendix III of the thesis.

7.2.2.1 Conclusion on system reliability assessment using β-unzipping method

The β-unzipping method was used as a method of system reliability assessment of a truss structure with two degrees of redundancy. A failure tree representation of the whole method that was applied to the structure provides a good perspective over the whole methodology. This is shown in Figures 7.23 and 7.24 for both cases of normal and non-normal random variables.

The failure trees shown in Figures 7.23 and 7.24 depict the functionality of the method. As is depicted in the figures, the β-unzipping method is a way to identify the important failure modes of the structure. Comparing the number of cut-set events identified through this method with a failure tree based on exhaustive enumeration of the failure modes proves that many of the other possible cut-set events were left out from the system reliability evaluation. This makes the system reliability evaluation less time-consuming, and provides a conservative estimation for the system reliability. It is also notable that the accuracy of the system reliability calculations through this method can be changed by adjusting the $\Delta\beta$ values chosen as the unzipping criterion. For instance, an increase in $\Delta\beta$ will lead to the inclusion of more branches of the failure tree (cut-set events), hence a more accurate estimation of the system reliability. As result, β-unzipping method has the advantage of inclusion of a measure of accuracy of the system reliability evaluation.

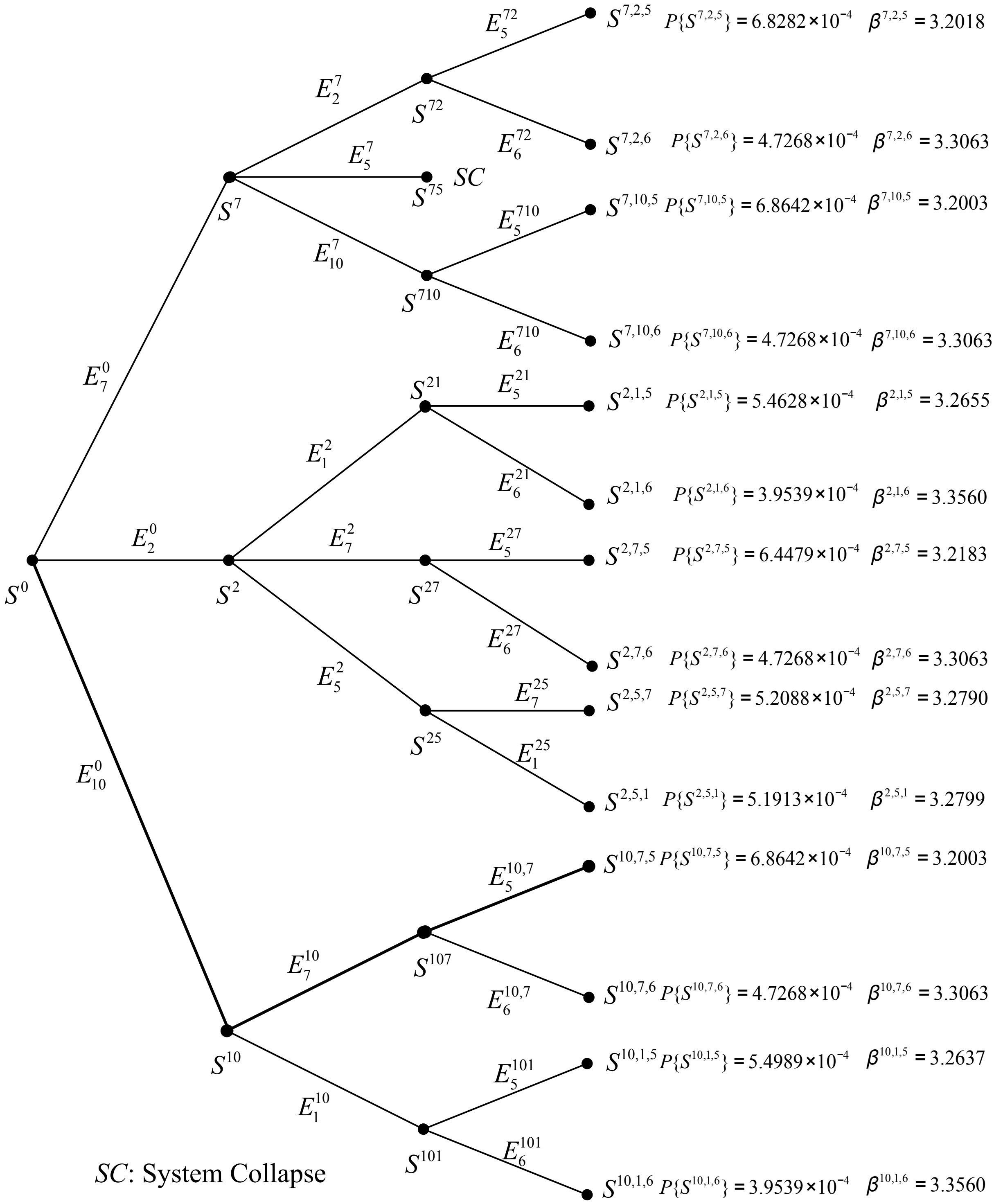

Figure 7.23: Failure tree for the truss structure Normal random variables

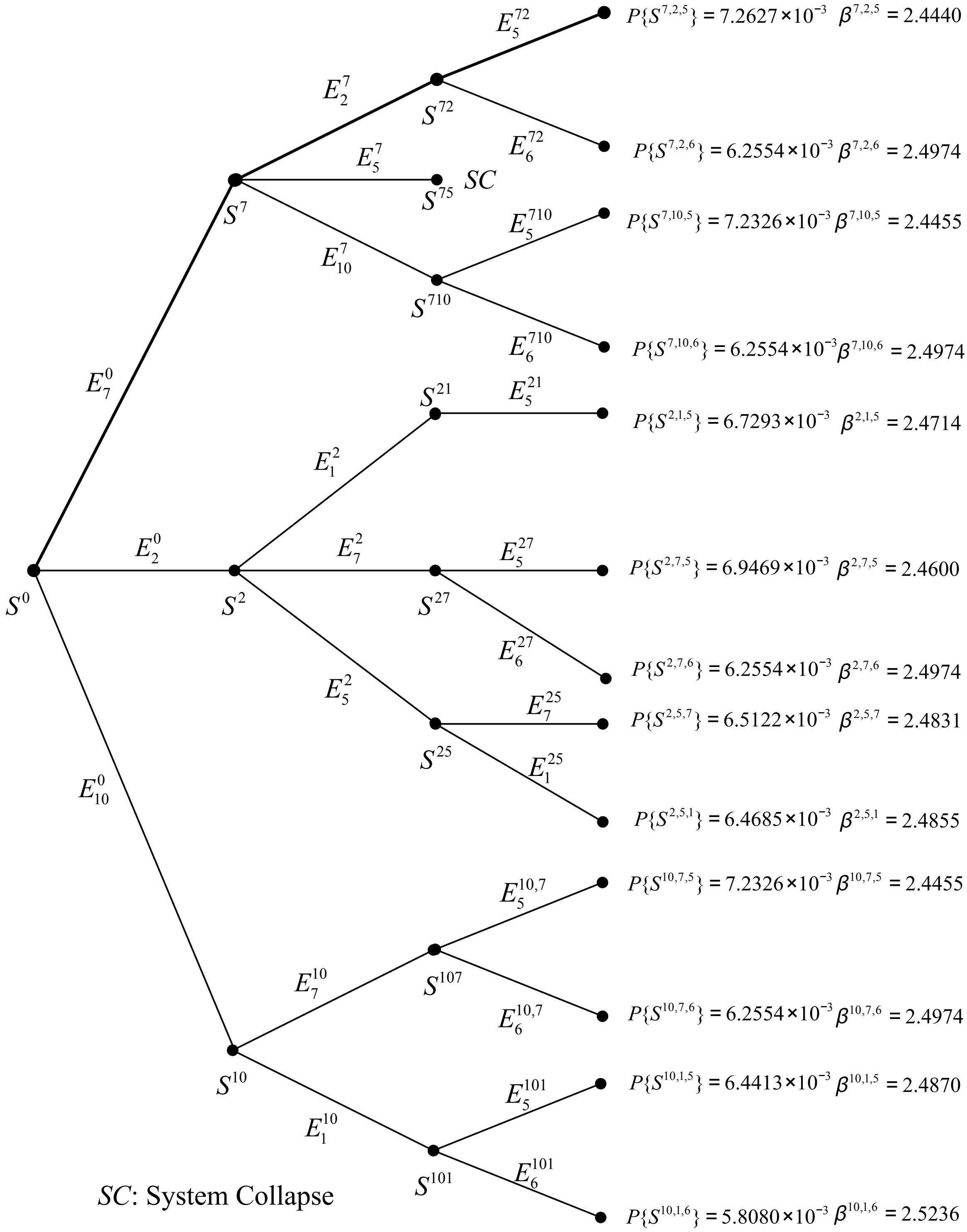

Figure 7.24: Failure tree for the truss structure Non-normal random variables

7.2.3 Computerised system reliability analysis of trusses based on β-unzipping method

In the previous sections the system reliability of a ten-bar truss structure was calculated using the β-unzipping method. This method uses the calculation of reliability index values at different damaged states of the structure to generate the most important failure modes of the structure. It was shown in Figures 7.23 and 7.24 that based on the selected unzipping intervals the different branches of the failure tree will be included or excluded from the reliability analysis. However, even selecting small unzipping intervals for the generation of the cut-set events (failure modes) will need a considerable amount of calculation. As a result, it is required to develop a computer program for system reliability evaluation using β-unzipping method.

In order to develop a computerised method for the system reliability analysis using the β-unzipping method four major modules need to be developed. These modules are: the general intact structure module, damaged structure module, reliability analysis module, and system reliability analysis module. For the purpose of obtaining the system reliability (or system reliability bounds), it is required that these modules be linked to each other. In other words, data from one module should be exported to another module where the module receiving the data will use it to perform its assigned task. Clarifications regarding these modules are given below:

1. **General intact structure module:** The general intact structure module is where the undamaged structure is defined. This module acts as a central module. Firstly, it encompasses all the information with respect to the geometrical definition of the structure, number of nodes, number of elements, member resistances, boundary conditions, orientation of the members, applied forces, etc. Secondly, it acts as a medium between different modules; hence, receiving and sending data to and from other modules. Finally, the module performs a FEM analysis of the structure, defines the critical elements based on the provided unzipping criteria, determining the system reliability bounds, and correlation matrix between the equivalent safety margins of the elements of the parallel-series system.

2. **Damaged structure module:** The damaged structure module is where the damaged state of the structure is modelled. This module receives all the required data form the general intact structure module. This is then used to model a certain damaged state of the structure. In fact, the damaged module is where the axial hinges are inserted, and the failed member is replaced by its post-failure capacity. This module should have the ability to model the axial hinge based on the failure type and failure mode of the failure element. The type of the failure can be ductile, semi-brittle, or brittle, and the failure mode can either be in tension or compression. The failure type can be accommodated by using a proper post-failure coefficient (for instance, C=1 for ductile failure, C=0.5 for a semi-brittle failure, C=0 for a brittle failure). Binary coding can be utilised for defining the failure type of the member (for example, 1 can represent failure in tension and 0 can represents failure in compression). The axial hinge is modelled by simply removing the

member from the structure, and replacing it with its post-failure capacity which can be performed using the following applied load vectors:

Failure in tension for brace elements with a positive orientation or cord members:

$$F_R = \begin{bmatrix} \cos(\theta) & \sin(\theta) & -\cos(\theta) & -\sin(\theta) \end{bmatrix}$$

Failure in tension for brace members with negative orientation:

$$F_R = \begin{bmatrix} -\cos(\theta) & -\sin(\theta) & \cos(\theta) & \sin(\theta) \end{bmatrix}$$

Failure in compression for brace members with a positive orientation or cord members:

$$F_R = \begin{bmatrix} -\cos(\theta) & -\sin(\theta) & \cos(\theta) & \sin(\theta) \end{bmatrix}$$

Failure in compression for brace members with a negative orientation:

$$F_R = \begin{bmatrix} \cos(\theta) & \sin(\theta) & -\cos(\theta) & -\sin(\theta) \end{bmatrix}$$

Here, a brace member with positive orientation is a component which makes a positive angle with the global x-axis, and a cord member is a member that has an angle of zero with respect to the global x-axis where θ is the angle of a component with respect to the global x-axis. The first element of the vector is the magnitude of the force applied in x direction in the first node of the failed member and the second element is the force that is applied in Y direction on the first node of the failed member. The third and fourth elements of the vector are the forces that are applied in X and Y direction on the second node of the failed component respectively. It should be noted that these vectors are in fact the X and Y components of a unit force applied in the direction of a member. The influence factors obtained in this manner can then be multiplied by the post-failure capacity of the failed member. The data generated through this module, which includes the influence factors with respect to the applied loads and the fictitious loads representing the post-failure capacity of the failed member, is sent to the reliability analysis module so that the reliability of the remaining components can be obtained.

3. **Reliability analysis module:** The reliability analysis module takes the data form either the general intact structure module or the damaged structure module, and performs a component level system reliability analysis. The reliability analysis is performed using the FORM reliability analysis method.

4. **System reliability module:** The system reliability module is utilised to calculate the system reliability of the parallel systems that are composed of critical elements. This module takes the critical pairs (or triples,) and the data regarding the elements composing

these pairs such as the reliability index values and the sensitivity coefficients. Consequently, this data is used to obtain the reliability of each parallel system as well as its equivalent safety margin (equivalent sensitivity factors).

A flowchart of the different modules is shown in Figure 7.25.

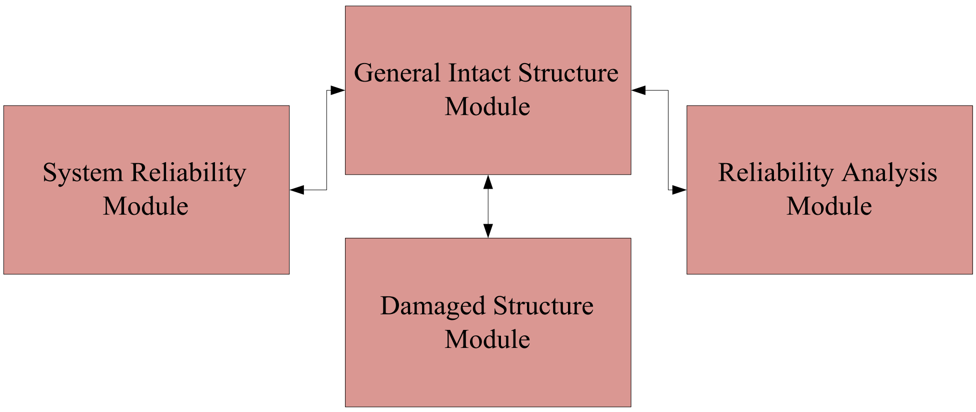

Figure 7.25: Flowchart for a computerised system reliability analysis using the β-unzipping method

7.2.4 Developed program for the analysis

In this section the computer program that was developed for the structural analysis of the truss structure is presented in detail. Required modules that are needed for the system reliability analysis were created. It enables the β-unzipping based system reliability calculation of the structure up to level 3. The programming code is provided in the Appendix II of the thesis.

7.2.4.1 The general intact structure module

The general intact structure is where the whole geometry of the structure, applied loads, and member resistances are defined. This module acts as a central module where all the other modules are linked together and the data received from them are processed. An outline of the module is presented algorithmically below:

1. Define the structure geometry, applied loads, and resistances (input basic random variables).

2. Perform a Finite Element analysis of the intact structure based on the data provided in the step 1.

3. Start the system reliability analysis at level 1 by sending the influence coefficients and member resistance in compression and tension to the reliability analysis module.

4. Input the value of $\Delta\beta_1$ for the determination of critical elements.

5. Retrieve the data from reliability analysis module.
6. Define the minimum reliability index.
7. Form the β-unzipping interval for level 1 based on the data provided in steps 5, 4 and 6.
8. Determine the critical failure elements based on the data in steps 5 and 7.
9. Form the correlation coefficient matrix for the critical elements using the data provided in sections 5 and 8.
10. Define the Ditlevsen bounds for the series system.
11. Start the system reliability analysis at level 2 by taking all the required information regarding the critical elements and their properties such as failure mode, orientation, etc.
12. Send the data retrieved in step 11 to the damaged structure module.
13. Receive the data from the damaged structure module for all of the possible damaged states of the structure.
14. Define the minimum reliability index for each possible damaged state of the structure.
15. Input the value of $\Delta\beta_2$ for the determination of critical element for all of the possible damaged states of the structure.
16. Determine the critical failure elements for all of the identified damaged states of the structure.
17. Identify the pairs of critical failure elements using the data provided in steps 16 and 8.
18. Send the identified pairs of critical elements and their properties such as sensitivity factors and reliability indices to the system reliability analysis module.
19. Form the correlation matrix for the parallel-series system through the data received from the system reliability module.
20. Form the Ditlevsen bounds for the system using the data retrieved form the system reliability module.
21. Start the system reliability at level 3 by taking all the information regarding the critical pairs of failure element.
22. Send the data from step 21 to the damaged structural module.
23. Receive the data from the damaged structure module from the analysis on all the possible damaged states of the structure.
24. Determine the minimum reliability index for each possible damaged state of the structure.

25. Input the value of $\Delta\beta_3$ and determine the β-unzipping interval for all of the possible damaged states of the structure.

26. Determine the critical failure element(s) for all of the possible damaged states of the structure at level 3 based on the data provided in steps 24 and 26.

27. Form the parallel triples of failure elements using the data provided in steps 26 and 17.

28. Send the data regarding the parallel triples to the system reliability module for level 3.

29. Form the correlation matrix for the parallel-series system through the data received from the system reliability module.

30. Form the Ditlevsen bounds for the system reliability evaluation at level 3 using the data retrieved form system reliability module.

7.2.4.2 Damaged structure module

The damaged structure module is used to model the different damaged states of the structure in the process of generating and identifying the important failure paths. It inserts the plastic hinges by removing the critical member(s) and replacing the critical member(s) with its post-failure capacity. In the program developed for the structure the assumption was that the failure of all the elements is ductile. Therefore, the failure type coefficient is not accommodated in the program. However, as was mentioned in section 7.2.3, this can be easily included in the program. An outline of the algorithm for the program is given below.

1. Receive all the data regarding the critical failure elements, their modes of failure, resistances, orientation, etc.

2. Identify the degrees of freedom numbers for the start and end node of the critical element.

3. Remove all the information regarding the critical failure element(s) form the vectors and matrices that are used to define the structure.

4. Form the vector of fictitious loads replacing the critical failure element(s) based on the orientation and failure mode of the critical element(s) using the load vectors defined in section 7.2.3.

5. Perform a FEM analyses for the load vector of fictitious loads as well as the load vector of imposed loads, and obtain the influence coefficients corresponding to each vector (separate FEM analyses are performed for each load vector).

6. Multiply the vector of influence coefficients corresponding to the fictitious loads by the proper resistance of the critical member(s) based on the failure mode of the failed member(s).

7. Form the new safety margins by obtaining the proper coefficients of resistance and load effect for the reliability analysis. These safety margins are the safety margins for both tension and compression failure of the remaining elements.

8. Send the data to the reliability analysis module.

9. Retrieve the data from the reliability analysis module, and identify the proper failure modes.

10. Extract all the required information such as reliability indices, modes of failure, and sensitivity factors.

7.2.4.3 System reliability analysis module

The system reliability module computes the system reliability of the parallel systems. It also finds the equivalent sensitivity factors for each parallel system. Clearly, with the equivalent sensitivity factors and the reliability index of the parallel system the equivalent safety margin can be formed. This module is only used for level 2 and 3 since in level 1 there are no parallel systems. An algorithmic outline of this module is given below.

1. Take the complete data regarding the critical elements which are forming the parallel system (parallel pair, parallel triple, etc.) such as reliability index values and sensitivity factors.

2. Form the correlation matrix which represents the correlation between the safety margins of the elements of the parallel system.

3. Find the reliability index of the parallel system using the correlation coefficient matrix and the reliability indices of the failure elements which are forming the parallel system.

4. Calculate the equivalent sensitivity factors for the parallel system based on the data obtained in steps 1 and 3.

7.2.4.4 Reliability analysis module

The reliability analysis module performs a component level reliability analysis. This module is used to evaluate the reliability of the safety margins in either the intact or damaged structure. The evaluation is based on the FORM reliability analysis method described in Chapter 3. The algorithm regarding this method was comprehensively presented in Chapter 5. As a result, it is not presented here.

It was shown in Section 7.2.3 that some of the safety margins are not applicable. It was noticed that these safety margins cause the divergence of the FORM algorithm or cause a “NaN” error (NaN : not a number) in Matlab. These errors are identified and the reliability index of these inapplicable safety margins are artificially chosen as a high value so that they are ignored in the 9th step of the damaged module.

This error is caused in cases where the load effect coefficient of b in the limit state equation ($G = a \times R - b \times E$) is considerably smaller that the resistance coefficient a (for instance, if $a = 1.2$ and $b = 0.2$). The "NaN" error is caused in the calculation of the equivalent normal parameters at the design point for the load effect. In this case in the process of the FORM analysis the value of the cumulative distribution function of the Gumbel distribution tends to 1 (refer to Equation 3.23) where the value of $\Phi^{-1}(F_X(x^*))$ tends to infinity. This will in turn cause the value of Equation 3.25 to tend to zero which means Equation 3.26 is the multiplication of zero by infinity ($0 \times \infty$) where this expression is mathematically of indeterminate form, hence causing Matlab to generate a "Not a Number (NaN)" error.

7.3 Branch and Bound method for system reliability

Another method that can be used for the system reliability evaluation of the structure is the branch and bound method that was mentioned in Chapter 6. This method is used to generate the failure paths of the structure or the cut-set events. These cut-set events are the same as parallel systems that were identified using the β-unzipping method. This search method is based on the formation and expansion of the structural failure tree by finding the nodes with high probabilities of failure or low reliability indices.

In order to apply the branch and bound method the failure tree of the intact structure needs to be formed. This way, it is possible to find the first node that needs to be changed to an internal node which is the node with the smallest reliability index or the node with the biggest failure probability. The rest of the nodes in the failure tree will remain as external nodes. Then, from the first internal node the failure tree is branched out. It means the damaged state is modelled and the reliability indices (probabilities of failure) of the generated damaged states are calculated (using the reliability of the remaining members and the failed member). This means that a new set of external nodes are introduced to the failure tree. Next, the second candidate to become an internal node needs to be identified. This is chosen as the node with the smallest reliability index among all of the external nodes in the failure tree. This node is either one of the newly generated external nodes or one of the remaining external nodes in the branches of the intact structural failure tree. This process is repeated, and each time a new internal node is identified. This will lead to the generation of important failure sequences or cut-set events. It should be noted that a cut-set event is formed when a terminal node is reached where a terminal node is any node that corresponds to the failure of the structural system. In this case, due to the degree of redundancy of the structure, a terminal node is a node representing the failure of three elements. In other terms, the failure tree is expanded up to a point where a terminal node is reached. These identified cut-set events can then be used to form a parallel-series model of the structure. The Ditlevsen bounds can be applied to these parallel-series system as was done for the β-unzipping method.

In this section, the branch and bound search method is applied to the case of non-normal random variables. A program is developed based on the available modules that were developed for the β-unzipping method to model the different damaged states of the structure. Accordingly, the results of this program are used to form the structural failure trees for various damaged states of the structure. However, the branch and bound search, which is used to identify the important failure sequences, is performed manually.

Through the branch and bound search, it is tried to find the same number of important failure modes (failure sequences) for the truss structure as with the β-unzipping method. The failure trees that are needed for the search to find external nodes are shown. Finally, the generated parallel-series system is compared with the one that was generated using the β-unzipping method, and based on the parallel-series model the Ditlevsen bounds will be formed for the system reliability.

7.3.1 Identifying the first internal node

The search is performed to find the first internal node. This node is found through the reliability evaluation of the intact structure. As is seen, the first internal node is the node representing the failure state S^7. This node is shown with number 1 in Figure 7.26. The other external nodes that are numbered in Figure 7.26 are found through the next steps of the search. Therefore, at this stage they are treated as external nodes.

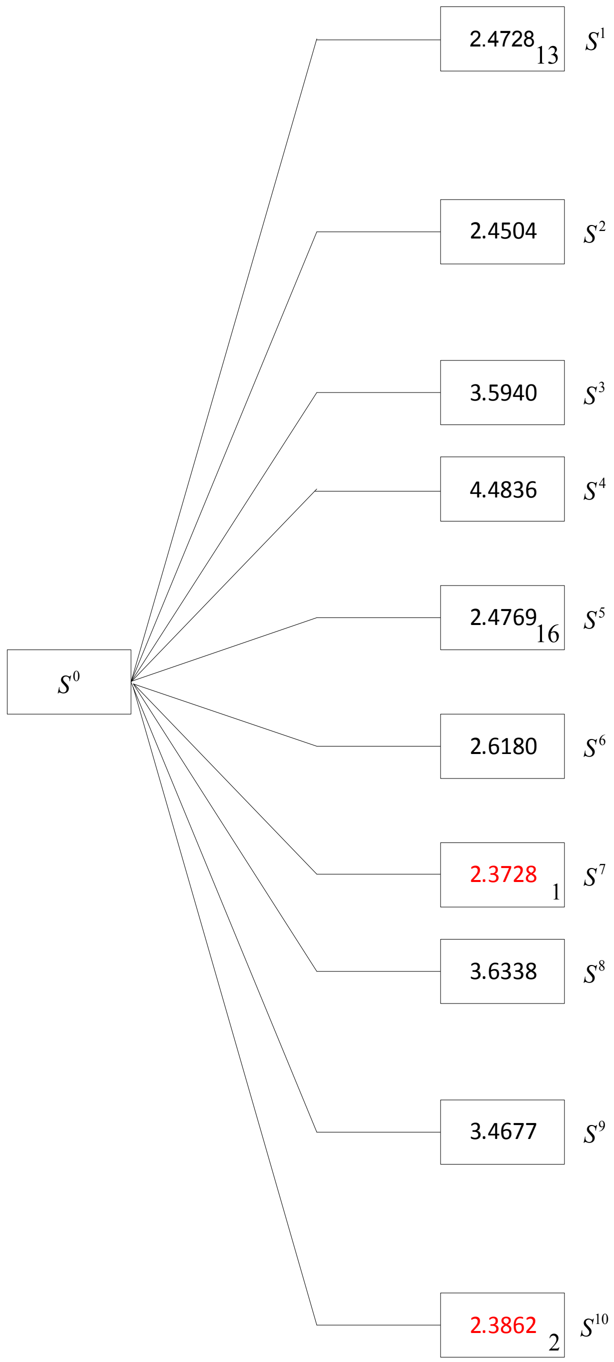

Figure 7.26: Failure tree for the intact structure showing RI values

The failure tree is now branched out from node S^7 (The first internal node). This leads to a new set of external nodes. Figure 7.27 shows the expansion of the failure tree that is branched out from node S^7. The second internal node is chosen from all of the external nodes depicted in Figure 7.27.

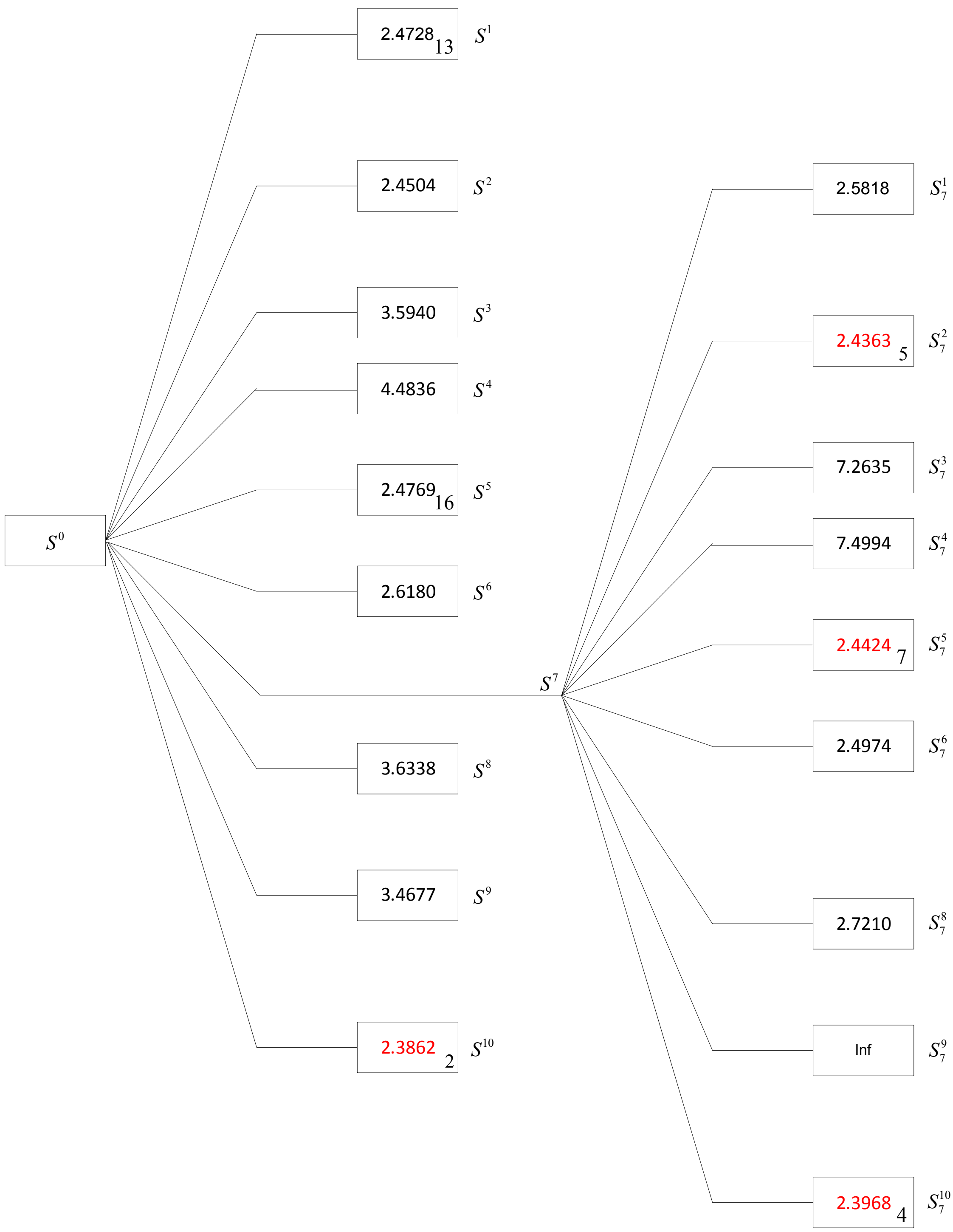

Figure 7.27: Failure tree for the damaged state S^7 showing RI values

Looking at the internal nodes, it is seen that node S^{10} in the failure tree of the undamaged structure has the smallest reliability index; hence this node is chosen as the second internal node. In Figure 7.26 this node is marked with number 2. This means the next damaged state to be modelled is the damaged state S^{10}.

Figure 7.28 shows the failure tree for the damaged state where member 10 has failed. This introduces 9 more internal nodes to the whole failure tree. The next internal node is chosen among all of the external nodes in the failure tree of the structure. The remaining number of external nodes at this stage of the search is 16.

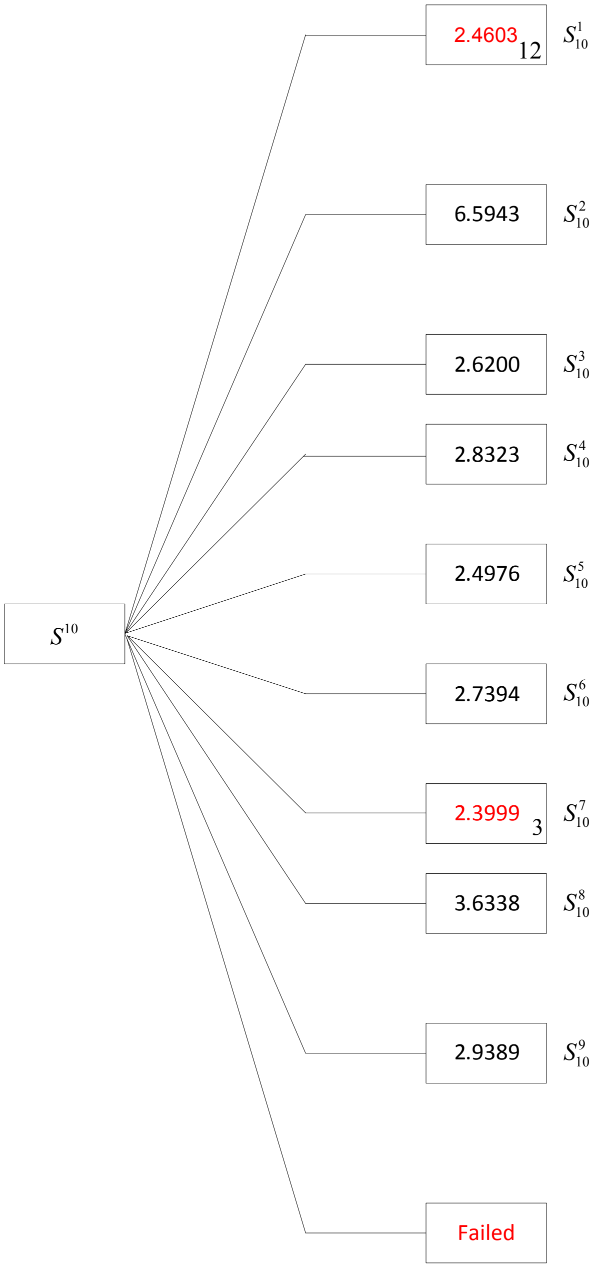

Figure 7.28: Failure tree for the damage state S^{10} showing RI values

The third internal node among all of the external nodes is node S_{10}^{7} since it has the smallest reliability index among all the internal nodes. This is shown in Figure 7.28. As a result, the next damaged state that needs to be modelled is the damaged state where members 10 and 7 have failed which corresponds to the third identified internal node. The failure tree for this damaged state is shown in Figure 7.29.

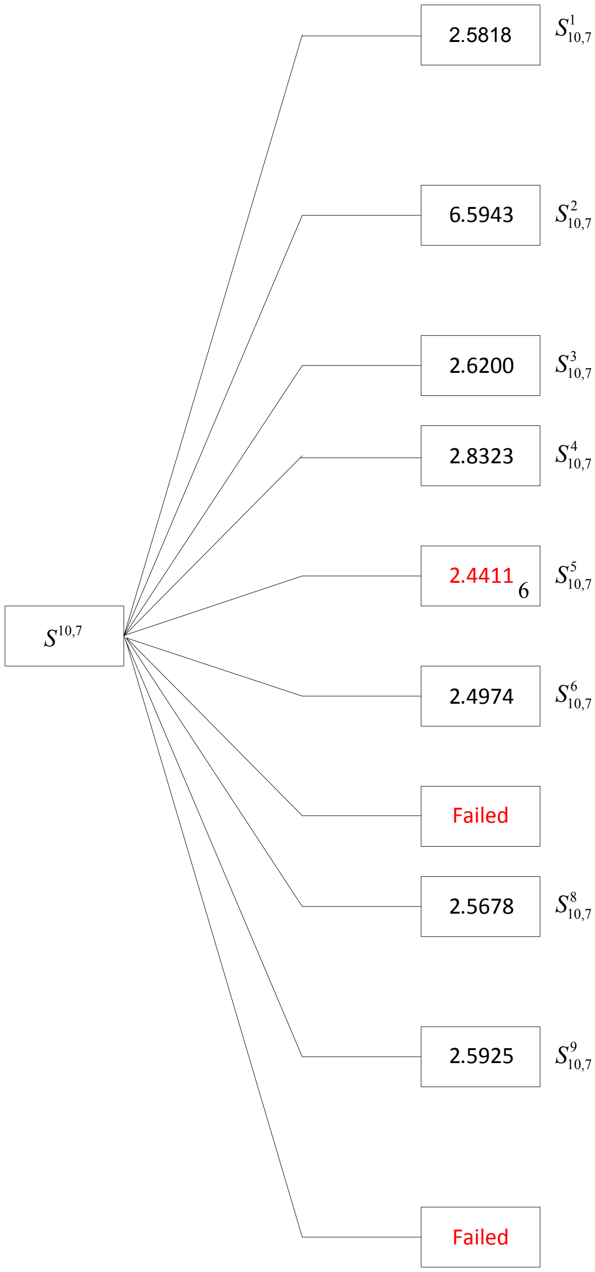

Figure 7.29: Failure tree for the damaged state $S^{10,7}$ showing RI values

Among the new set of external nodes, node S_7^{10} has the smallest reliability index and is chosen as the fourth external node which is shown in Figure 7.27. Although, the damaged state is repetitive, it has to be considered due to the difference in sequences. It is shown in Figure 7.30.

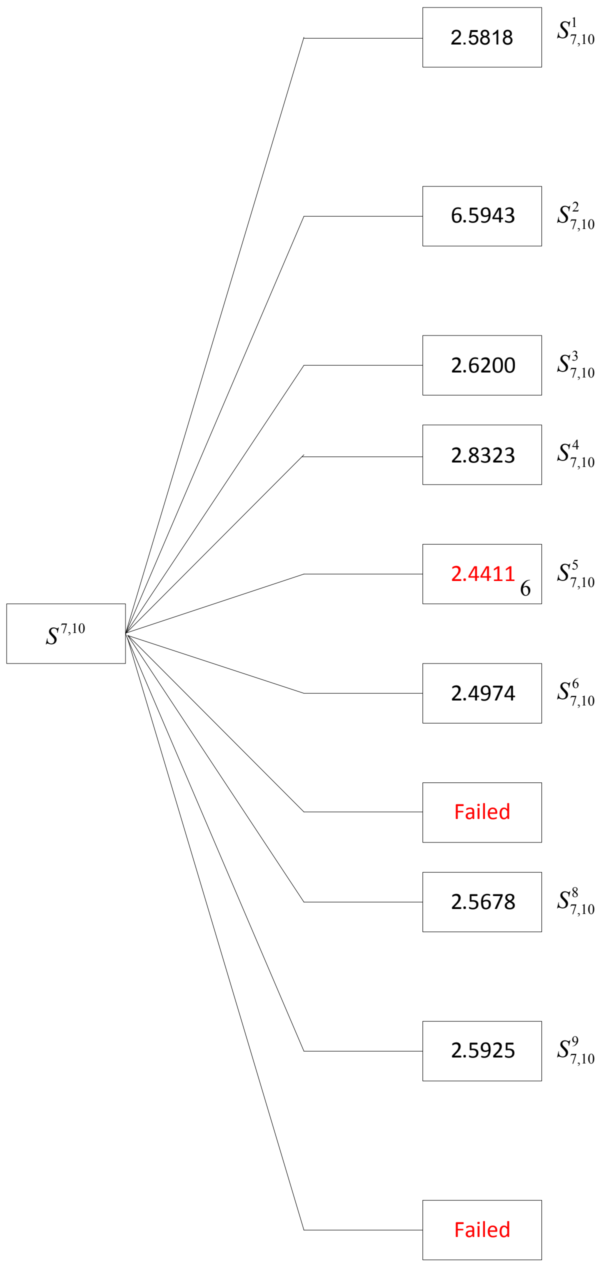

Figure 7.30: Failure tree for the damaged state $S^{7,10}$ showing RI values

The fifth external node is S_7^2. This is also shown in Figure 7.27 which shows the failure tree for the damaged state where member 7 is removed.

According to the fifth identified internal node, the next damaged state to be considered is the damaged state where elements 7 and 2 have failed.

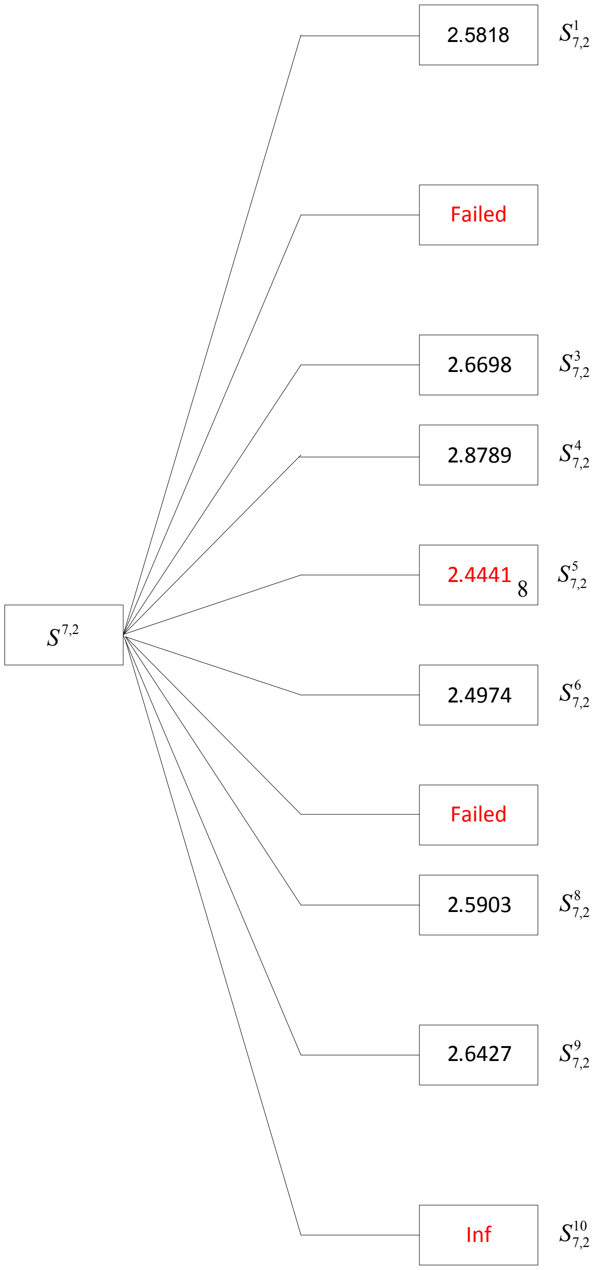

Figure 7.31: Failure tree for the damaged state $S^{7,2}$ showing RI values

Considering the new external nodes shown in Figure 7.31, the sixth internal node can be node $S^5_{10,7}$ as well as $S^5_{7,10}$ depicted in Figure 7.29 and 7.30, respectively. Both of the nodes are numbered as 6 since they posses the same reliability indices. This means the first and second failure sequences are the sequence composed of elements 10 & 7 & 5 and 7 & 10 & 5. Since the removal of more than two members causes the instability of the whole system, these nodes are considered as terminal nodes.

The seventh internal node is node S^5_7. This node is shown in Figure 7.27, and is marked with number 7. As a result, the damaged state where member 7 and 5 have failed needs to be modelled, but it was mentioned that this will turn the structure into a mechanism. Thus, this damaged state is ignored for now in that it doesn't form a parallel triple of failure elements. This issue will be addressed in section 7.4.

The eighth internal node is node $S^5_{7,2}$ shown in Figure 7.31. This means the third failure sequence or failure path is composed of element 7, 2, and 5.

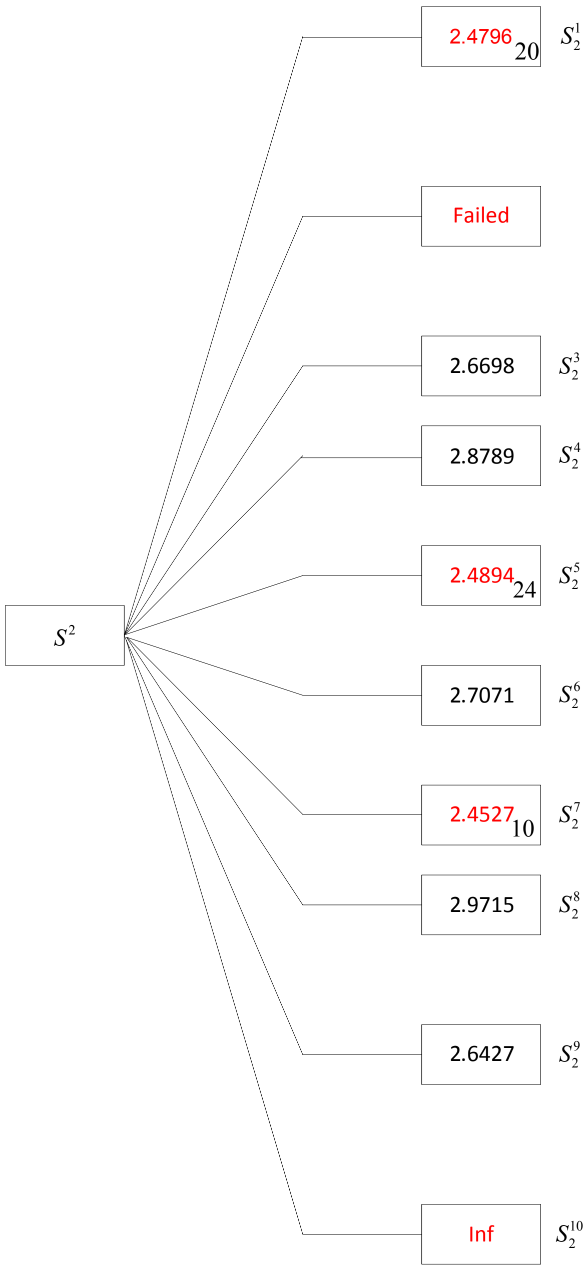

Figure 7.32: Failure tree for the damaged state S^2 showing RI values

The ninth internal node can then be determined as node S^2 since it has the smallest reliability index among all the remaining external nodes. Therefore, the damaged state where member 2 has failed needs to be modelled. This is shown in Figure 7.32.

The tenth internal node is node S^7_2. Although, this physical damaged state was already investigated, it needs to be considered due to the fact that it is obtained through a different sequence. The next node to become internal is node $S^5_{2,7}$ which is shown in figure 7.33. This node is marked with number 11.

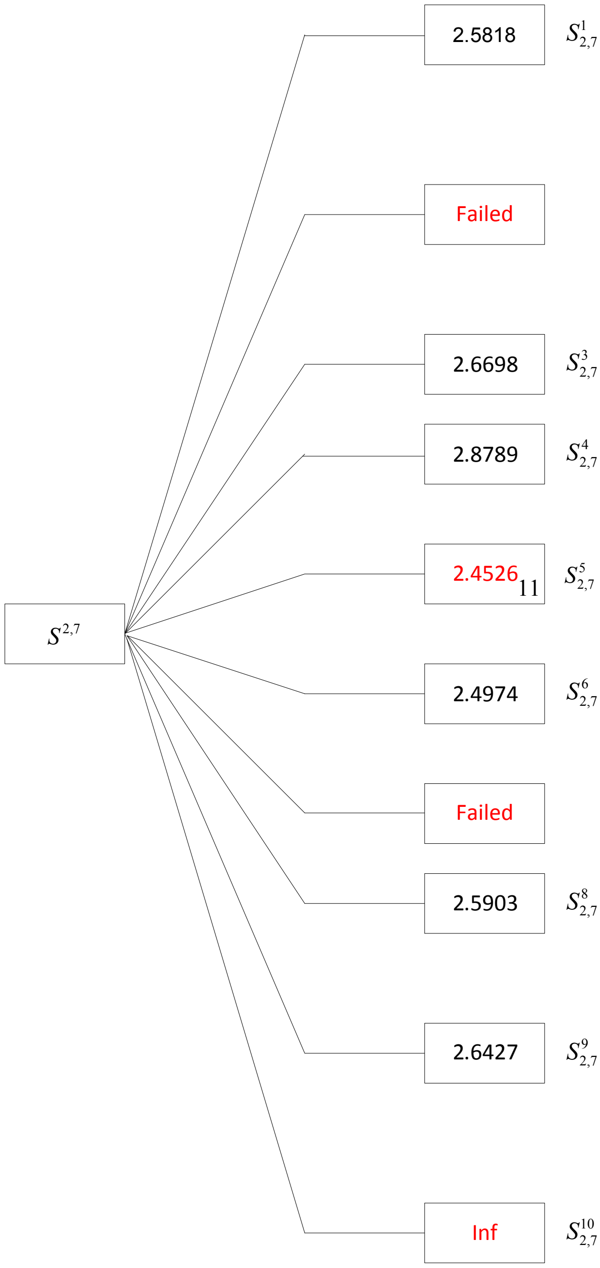

Figure 7.33: Failure tree for the damaged state $S^{2,7}$ showing RI values

Node $S^5_{2,7}$ is a terminal node; therefore, the fourth cut-set event that is obtained through the branch and bound method is the sequence of element 2 & 7 & 5.

The search for the twelfth internal node leads to node S^1_{10} shown in Figure 7.28. The damaged state where members 10 and 1 have failed is modelled. The failure tree for this damaged state is presented in Figure 7.34.

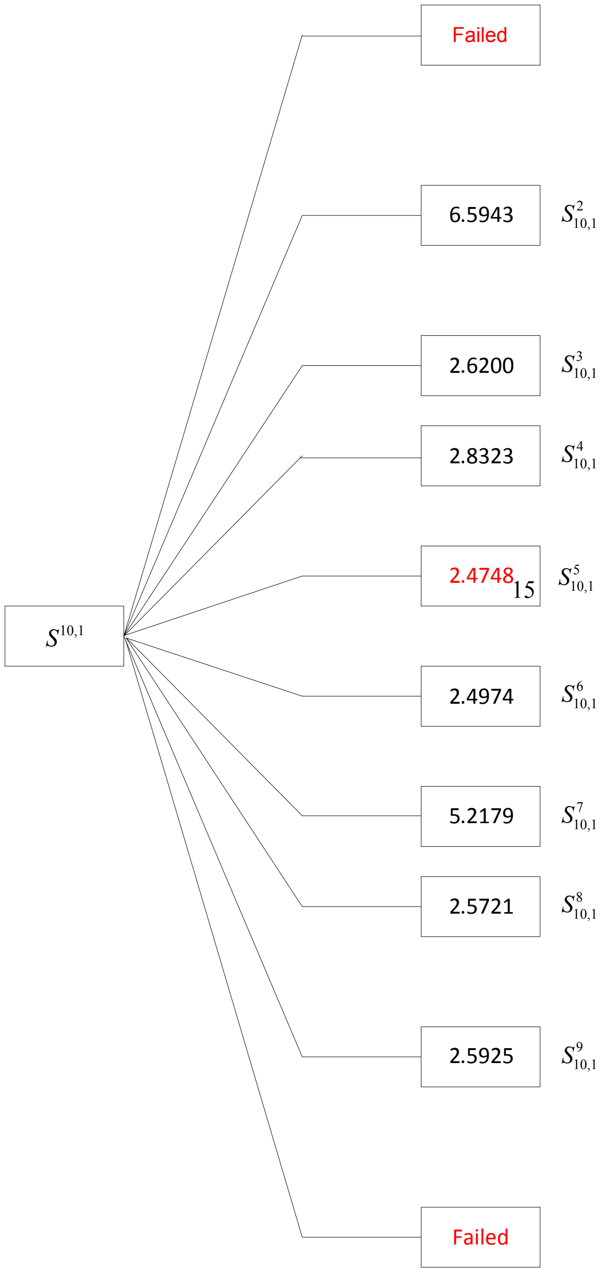

Figure 7.34: Failure tree for the damaged state $S^{10,1}$ showing RI values

The search for the thirteenth internal node yield nodes S^1. The damaged state where member 1 has failed is modelled. The failure tree for this damaged state of the structure is depicted in Figure 7.35.

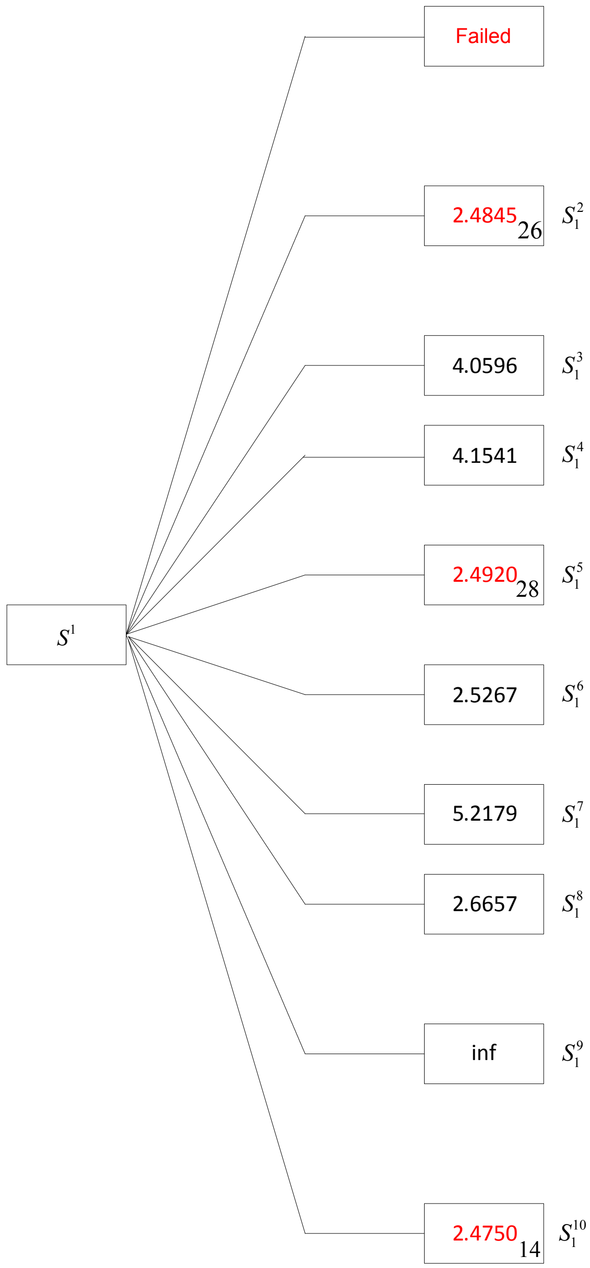

Figure 7.35: Failure tree for the damaged state S^1 showing RI values

As is seen in Figure 7.35 the next internal node is node S_1^{10}. This node is marked as number 14. As a result, the damaged state where members 1 and 10 have failed needs to be modelled. Previously, the damaged state for the sequence of elements 10 and 1 was modelled. Now, the sequence 1-10 needs to be investigated. This is shown in Figure 7.36 below.

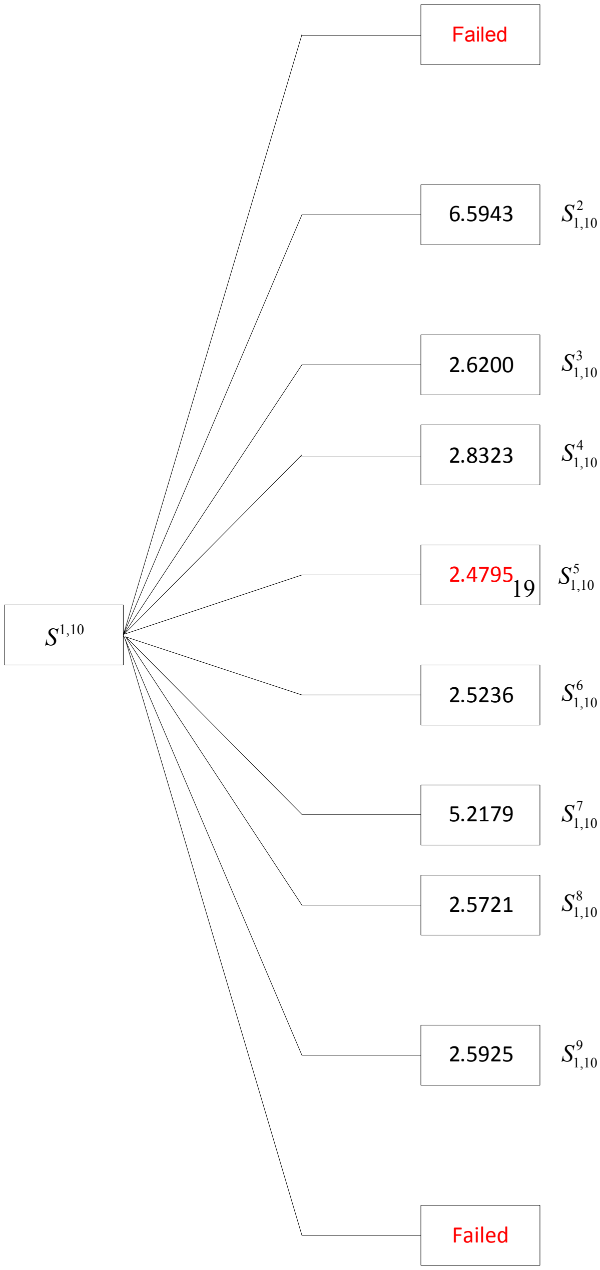

Figure 7.36: Failure tree for the damaged state $S^{1,10}$ showing RI values

The branch and bound search for the fifteenth external node leads to node $S^5_{10,1}$ shown in Figure 7.34. This means the fifth failure sequence that is found is the sequence of elements 10, 1, and 5.

The sixteenth internal node is node S^5 which is located in the failure tree of the intact structure shown in Figure 7.26. The damaged state needs to be modelled. The failure tree for this damaged state is shown in Figure 7.37.

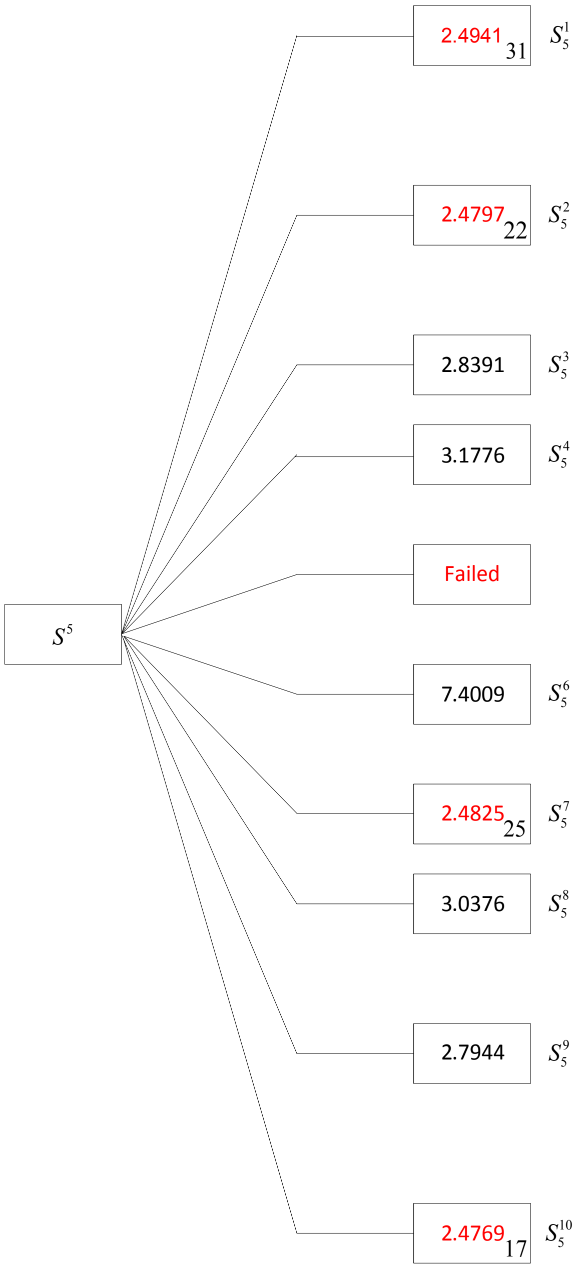

Figure 7.37: Failure tree for the damaged state S^5 showing RI values

The seventeenth internal node is shown in Figure 7.37. The failure tree for this node refers to the damaged state where members 5 and 10 have failed.

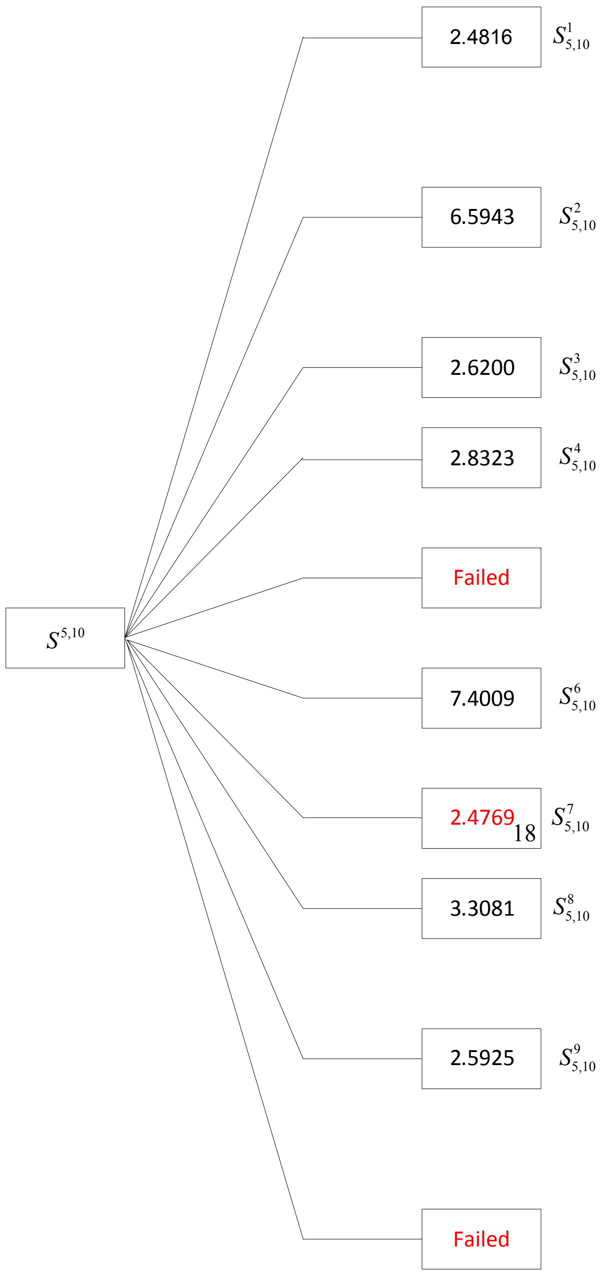

Figure 7.38: Failure tree for the damaged state $S^{5,10}$ showing RI values

The node referring to the failure state $S^{7}_{5,10}$ in this failure tree is the next node to be internal. This node is a terminal node. This means the sixth cut-set event is the sequence of elements 5 & 10 & 7.

The nineteenth external node is found to be node $S^{5}_{1,10}$ which is shown in Figure 7.36. Consequently, the seventh cut-set event is composed of elements 1 & 10 & 5. The 20th node is identified as node S^{1}_{2}. Therefore, the damaged state where members 2 and 1 have failed needs to be modelled. The failure tree is expanded for this damaged state which is shown in Figure 7.39.

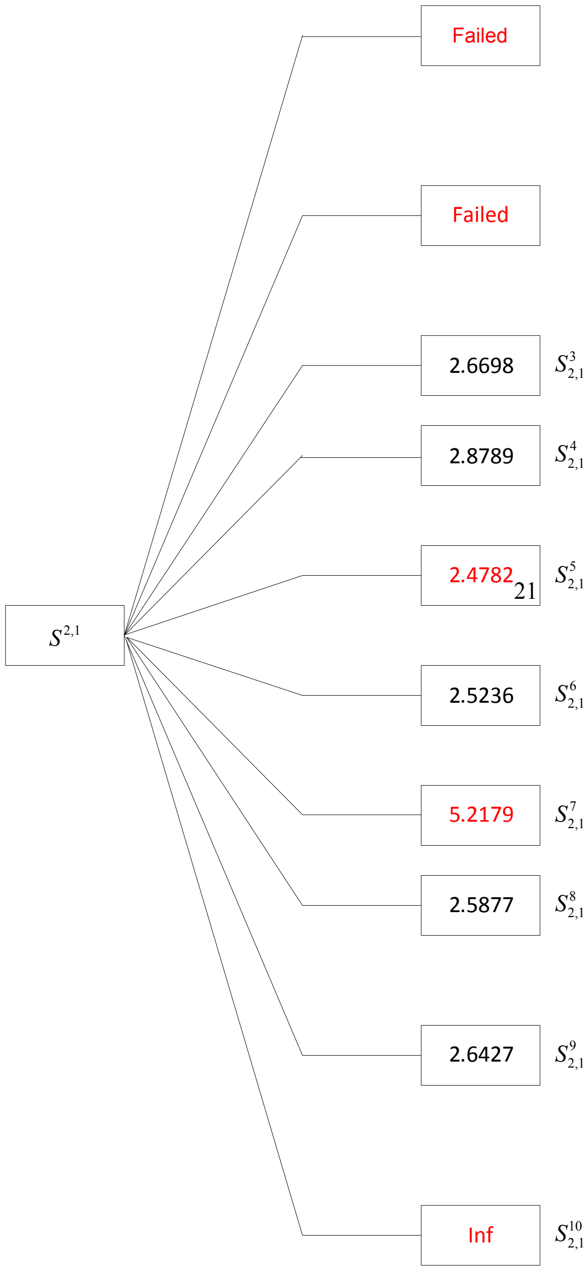

Figure 7.39: Failure tree for damaged state $S^{2,1}$ showing RI values

It is clear form Figure 7.39 that the 21^{st} internal node is $S_{2,1}^{5}$ which means the eighth identified failure sequence will be the sequence of elements 2, 1, and 5.

The 22^{nd} external node to be formed is node S_5^2. This node is shown in Figure 7.37 and is marked with number 22. This means the damaged state where elements 5 and 2 have failed needs to be modelled. The failure tree for this damaged state is shown in Figure 7.40.

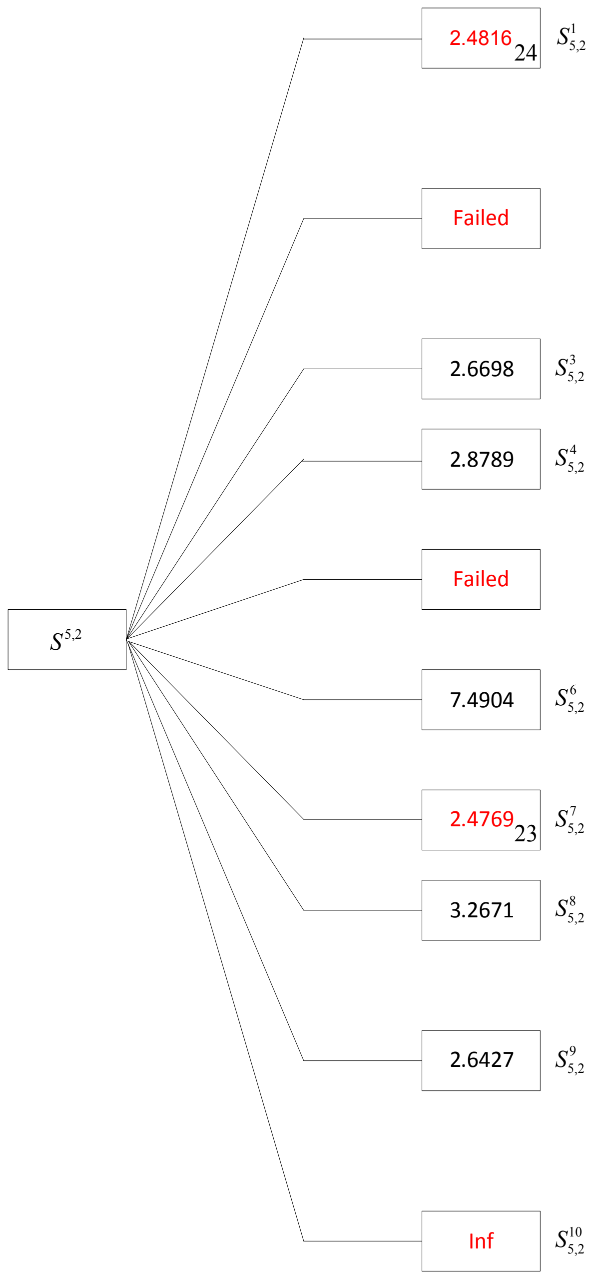

Figure 7.40: Failure tree for the damaged state $S^{5,2}$ showing RI values

The node referring to state $S^7_{5,2}$ is identified as the 23$^{\text{rd}}$ internal node in the branch and bound search, and node $S^7_{5,2}$ is identified as the 24$^{\text{th}}$ internal node. Both of these nodes are terminal nodes; therefore, the ninth and tenth cut-set events are identified.

The next node (node 25) in the branch and bound search is node S^7_5 shown in Figure 7.37. This node, however, causes the collapse of the system, and the structural failure tree cannot be branched out further following this path.

Node twenty-sixth refers to a the damaged state where members 1 and 2 have failed. This is depicted in Figure 7.41.

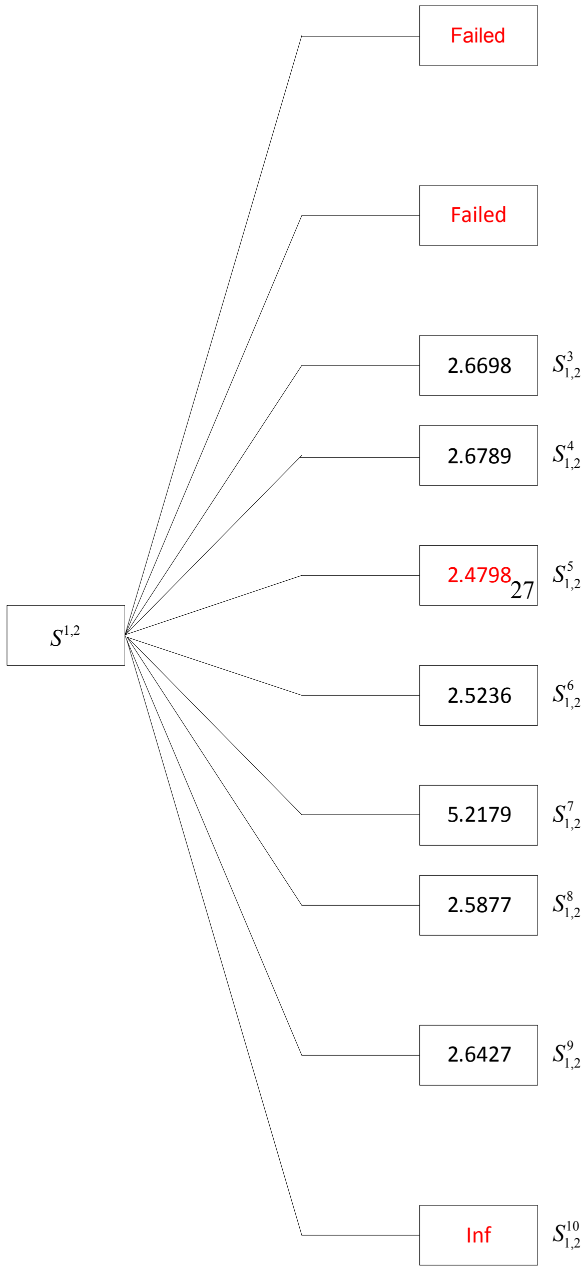

Figure 7.41: Failure tree for the damaged state $S^{1,2}$ showing RI values

It is seen in Figure 7.41 that the 27th node is the node $S^5_{1,2}$. This node is a terminal node, hence the 11th failure sequence is the sequence of elements 1 & 2 & 5. Next, node S^1_5 is chosen as the twenty-eighth node, which is depicted in Figure 7.42.

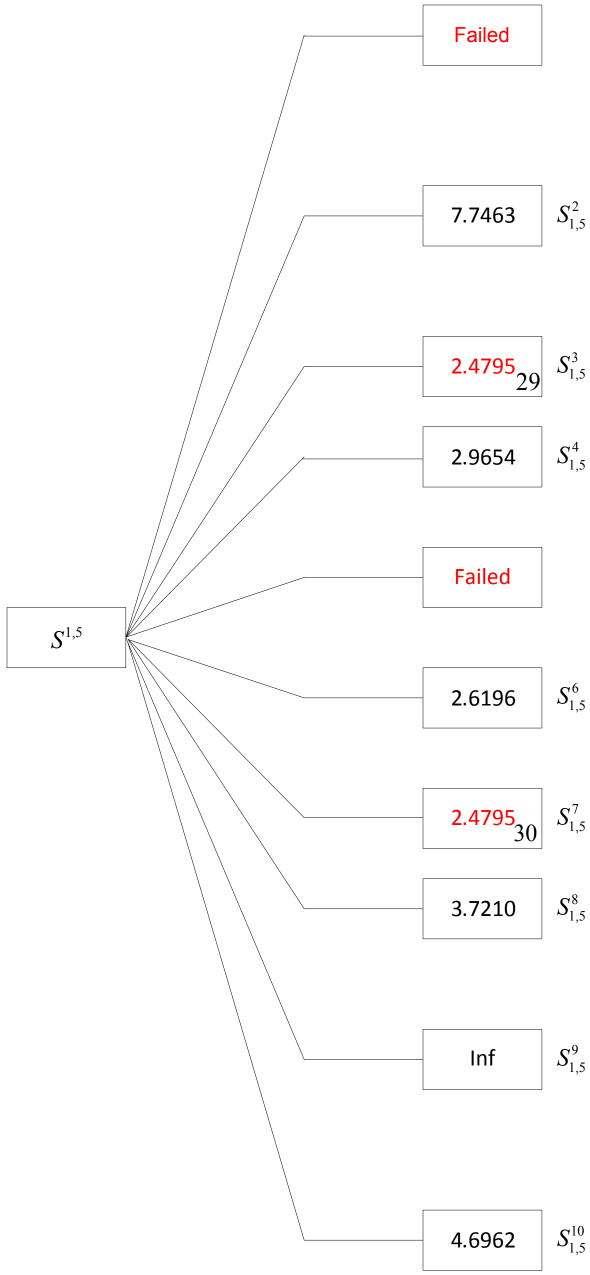

Figure 7.42: Failure tree for the damaged state $S^{1,5}$ showing RI values

The twenty-ninth and the thirtieth nodes are identified in the damaged state shown in Figure 7.42. These nodes represent terminal nodes; Therefore, failure sequences 1, 5, and 3 as well as 1, 5, and 7 are chosen as the 12^{th} and 13^{th} failure sequences.

Thirteen different failure sequences where identified. The search is terminated here as there are the same number of failure sequences identified as that of β-unzipping method. The search, however, can be continued further in order to identify more failure sequences where the next candidates to become internal nodes could be identified.

The parallel-series model for the structural system obtained through the branch and bound search method is shown in Figure 7.43. In the model defined for the system reliability the cut-set events that were identified form the parallel systems, and the combination of these cut-set events form the whole series system.

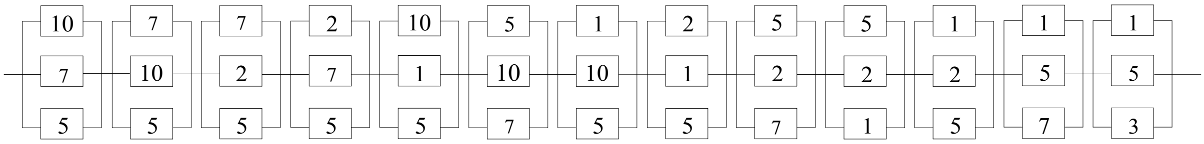

Figure 7.43: Parallel-series system obtained through the branch and bound search

The Ditlevsen Bounds can be formed for the system reliability of the series system whose elements are composed of parallel triples. This is shown below:

$$7.3211 \times 10^{-3} \leq P_{f_{sys}}^{BB} \leq 8.4831 \times 10^{-2}$$

It was shown through the beta-unzipping computation that the elements of the truss are highly correlated, so the lower Ditlevsen bound can provide an estimate of the system reliability.

$$P_{f_{sys}}^{BB} = 7.3211 \times 10^{-3} \rightarrow \beta_{sys}^{BB} = 2.4411$$

7.3.2 Conclusion on system reliability assessment using the branch and bound method

The system reliability of the truss structure was analysed using the branch and bound search method. Through this search method 13 most important failure paths were identified and accordingly, a parallel-series model was developed for the system reliability evaluation and the formation of the Ditlevsen bounds for the system.

In conclusion, the branch and bound search method is a probabilistic method that can effectively identify the failure sequences of the structure. The method is a robust method in a sense that it is completely based on the results of a stochastic analysis of the structure. The path of the search is only "navigated" by the probability of failure or reliability index values of different damaged states of the structure. Thus, all of the critical damaged states of the structure are considered. In other term, all of the nodes of the failure tree that refer to a damaged state with a high probability of failure are naturally included in the search, not to mention the fact that this itself leads to generating and identifying a larger number of critical nodes. Nevertheless, throughout the method, it was observed that based on the formation of internal nodes, it is necessary to shift between different levels of damaged states. For instance, it might be needed to shift from a damaged state at level 3 (failure of two elements) to a damaged state at level two (failure of only one element) or even to the undamaged state of the structure. Therefore, programming of the branch and bound search method seems relatively time-consuming and difficult. As a result, the calculation of the reliability indices were performed through a developed program, but the search of the failure paths was performed manually.

In Figure 7.44 a failure tree showing the identified failure sequences is presented. This failure tree only shows the identified internal nodes through the search, and is not depicting the whole failure tree of the structure.

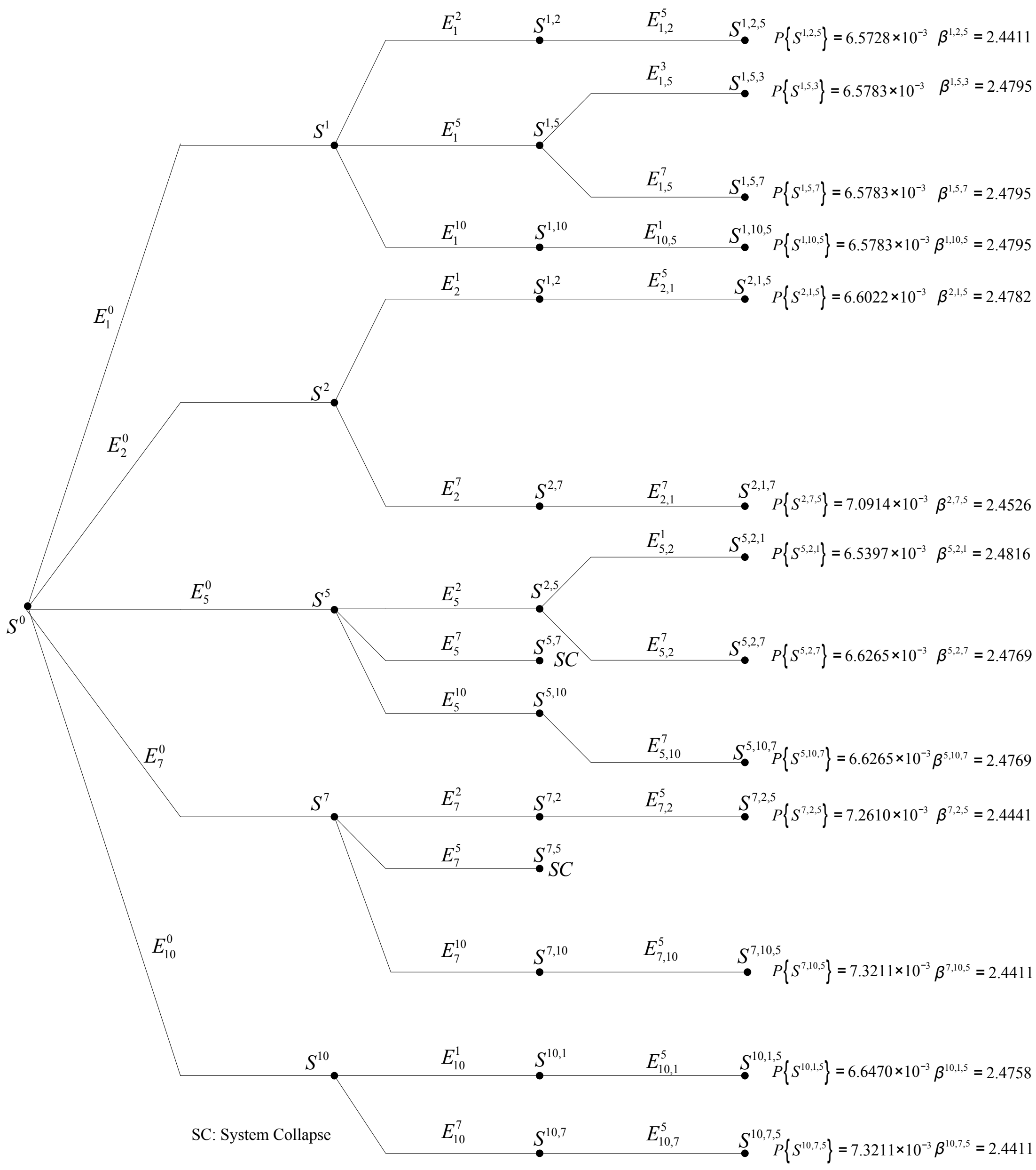

Figure 7.44: Failure tree for the branch and bound search method

7.4 Conclusion on system reliability analysis

The system reliability of a sample truss structure was investigated using two methods of system reliability analysis. First, the system reliability analysis was performed using the β-unzipping method for both of the cases of normal and non-normal random variables. Next, the branch and bound method was used to evaluate the system reliability of the truss structure only for the case of non-normal random variables. Both methods present ways of generating the failure sequences, and were used to create a parallel-series model for the whole structural system.

In general, through both methods, it was observed that there are certain elements that are more critical and play an important role in the system reliability of the structure. Accordingly, by looking at the failure elements forming the parallel-series model, it is seen that elements 7, 10, 5, 6, and 1 are quite influential in the system reliability of the structure. This fact is verified by both the β-unzipping method as well as the branch and bound approach since both of the methods are probabilistic methods that use the reliability evaluation results to search for the most important failure sequences. Nevertheless, other components of the structure are to some extent influential in the system reliability. The inclusion of less critical elements such as elements 8, 9, 3 and 4 can be accommodated by choosing higher unzipping intervals in the β-unzipping method or by further expansion of the structure failure tree in order to search for more failure paths in the branch and bound search approach.

Through comparing the different failure trees and parallel-series models developed for both of the methodologies, it is possible to pinpoint the differences and similarities of the branch and bound and the β-unzipping methods. Both of the aforementioned system reliability methods are similar in a sense that they are both probabilistic methods. The objective of both methodologies is to identify and generate the failure mechanisms (failure sequences or failure paths) based on a stochastic evaluation of the structure, where the same methodology can be used for the modelling and reliability evaluation of each damaged state of the system. Moreover, it is possible to use both of the methodologies to form a parallel-series model for the whole system as well as the same approach to form the reliability bounds for the system reliability assessment such as Ditlevsen bounds. Nevertheless, it is clear from the failure trees generated for the β-unzipping method and the branch and bound method that for the formation of similar number of failure sequences the two methods are slightly different. In the β-unzipping method the part of the failure tree branching out from failure states S^1 and S^5 were ignored whereas the failure tree resulted from the branch and bound search includes this failure states which leads to the generation of failure sequences 1-5-3, 1-2-5, 1-5-7,1-10-5 and 5-2-1, 5-2-7, 5-10-7, respectively. The failure states S^1 and S^5 were ignored in the β-unzipping method because the unzipping interval at level one was not large enough to include these damaged states. Conversely, in the branch and bound search, it is possible to go back to a different level of the system and generate different failure paths. In general the following distinctions can be made between these two methods:

1. The β-unzipping method is dependent on the level where the evaluation is performed, and once a certain damaged state is excluded from the failure path generation process, it is not possible to include it again. On the other hand, the branch and bound method shifts between different damaged levels in order to generate the failure paths.

2. The β-unzipping method is easier to computerise in that it follows a linear procedure of

failure path generation and damaged levels are evaluated in turn, but in the branch and bound method, it is necessary to shift between different levels which makes the method more difficult to program.

3. The β-unzipping method is highly dependent on the unzipping intervals where these intervals are selected completely arbitrarily. In contrast, the branch and bound method is entirely independent of any chosen interval and is solely based on the reliability evaluation results of different damaged states (nodes in the failure tree).

4. The failure path generation in the β-unzipping method is based on the definition of different damaged levels. For instance, the definition of system reliability evaluation at level 3 is the formation of a parallel-series model for the system which is composed of parallel triples of failure elements. The methodology is somewhat unclear regarding the failure mechanisms happening at an earlier level such as the failure sequence of elements 5 and 7 in the structure of Figure 7.1. This failure sequence corresponds to the system reliability evaluation at level two; however, due to the degree of redundancy of the structure, failure sequences at level 3 were also identified. A solution to this inconsistency can be the so-called system reliability evaluation at mechanism level. Nonetheless, this can make the computerisation of the method significantly complex.

   The branch and bound method is, however, based on the identification of terminal nodes where a terminal node is any node referring to the collapse of the system. It should be noted that for the sake of comparison the terminal nodes of states $S^{7,5}$ and $S^{5,7}$ were ignored in the system reliability evaluation in Section 7.3.1. If these failure mechanisms are also included in the system reliability analysis, the following parallel-series model can be used. This parallel series model is shown in Figure 7.45.

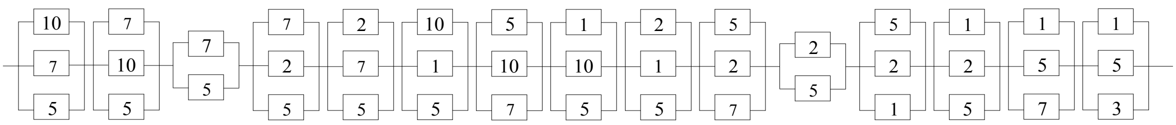

Figure 7.45: parallel-series model for the system reliability

As is seen in Figure 7.45, in the parallel-series model elements 5 and 7 are also included. The Ditlevsen bounds for this model is shown below where $P\{S^{7,5}\} = 7.2950 \times 10^{-3}$ and $P\{S^{5,7}\} = 6.5232 \times 10^{-3}$.

$$7.3211 \times 10^{-3} \leq P_{f_{sys}}^{BB} \leq 9.7433 \times 10^{-2}$$

The inclusion of the sequence of elements 5 and 7 doesn't change the lower bound of the Ditlevsen bounds for the system since the lower bound is the maximum of failure probabilities of the elements of the series system and the minimum belongs to the sequence of element 5,7,10. But it does change the upper bound of the system. It should also be noted that in this case the system reliability doesn't change since the lower bound of the system bounds didn't change (the system reliability is equal to the lower Ditlevsen bound due to high correlation between the safety margins of the structural elements in this case).

In short, both of the methods appear to be efficient in system reliability calculation. The β-unzipping method is easier to program, and this makes it easier to choose bigger intervals for the system reliability calculation which consequently makes it possible to include a bigger number of critical failure elements that might be ignored in the case of smaller intervals. Therefore, sufficient accuracy for the method can be provided. the branch and bound method could also be used for the system reliability, but the computer program developed for this method is only used to model different damaged states and the rest of the procedure is performed manually which can become rather complex. Therefore, it is recommended that a β-unzipping method with large intervals be used instead of the branch and bound method.

Chapter 8

Conclusions and Recommendations

8.1 General

Chapter 8 is dedicated to the conclusions and recommendations for the further study in this area of research. The thesis was mainly focused on a fully computerised stochastic evaluation of truss structures. Throughout this study common methods of component and system level reliability analysis were investigated where computer algorithms and programs were developed based on these methods. Clearly, it follows the objective of providing a completely stochastic finite element environment for a computerised system and component level evaluation of truss structures.

8.2 Conclusions and recommendations

Conclusions and recommendations are presented in the following sections regarding a component and system level reliability assessment of truss structures.

8.2.1 Component level reliability evaluation

Different methods can be used for a computerised component level reliability analysis of the truss structures or structures in general. Mostly, these methods are defined in three categories:

- Stochastic finite element formulation based on First Order Reliability Methods (FORM).
- Response surface method.
- Simulation methods.

Algorithmic methods were developed for the reliability evaluation using these methods and the following conclusions are presented below:

- All of the above-mentioned methods can be properly used for a computerised component level reliability analysis. However, in order to perform an accurate as well as a robust component level reliability evaluation proper interaction should be provided between the reliability analysis module and the finite element analysis. For a First Order Reliability Method (FORM), the deterministic finite element analysis is of paramount importance since it is used for the calculation of the derivatives of the performance function with respect to the basic input random variables. For simulation methods, the deterministic FEM analysis forms the basis for simulation. The number of failures of the performance function are counted based on the finite element analysis which leads to the final calculation of the probability of failure of the component. For the Response Surface Method (RSM) the results of the FEM analysis are utilised to form the database for the regression and formation of the component limit state functions. The reliability analysis is then performed on the obtained performance functions. As a result, all of the methodologies are highly dependent on the deterministic finite element analysis.

- The finite element analysis that is required for the reliability methodologies can be performed using a commercial finite element package where the Application Programming Interface (API) of the FEM package can be used to link the reliability analysis modules to the FEM analysis module. Another alternative is to develop an appropriate finite element coding in the same programming language as for the reliability analysis module development. In general, however, the latter is recommended since the interaction between the FEM program and the reliability analysis module can be time-consuming. Moreover, programming using an application programming interface can be rather time-consuming and complex on its own. Plus, this problem makes it impossible to use simulation methods (due to it being too slow) and restricts the reliability evaluation only to the first order reliability methods. As a result, the former should only be used where finite element programming of the structure is more complex than developing a proper API programming.

- Among the proposed simulation methods Updated Latin Hypercube Sampling (ULHCS) appears to be the most efficient method. However, ULHCS can be somewhat slower than Latin Hypercube Sampling Monte Carlo (LHCSMC) or Direct Monte Carlo Simulation (DMCS). This is due the fact that the updating process requires the rearrangement of the permutation matrix. if a faster simulation method is required, either LHCSMC or DMCS can be used.

- The response surface method is the most difficult approach for a completely computerised component level reliability analysis. This is owing to the fact that the whole process of generating the required information for the regression as well as the formation of the limit state functions is not as straight forward to program as the other two methods. For the case where strength performance functions are being investigated through linear elastic finite element analysis, the relationship between the internal member forces and the external forces can be established by application of an external vector of unit loads replacing the actual values of the external forces. This is achieved because the relationship between the applied forces and the displacements are linear. The usefulness of this method is employed in the system reliability evaluation of the truss structures.

- When the simulation methods are used for the component level reliability evaluation and all the basic input random variables are normally distributed, it is considerably more efficient to use the data provided through the simulation process for the calculation of the mean and standard deviation of the performance functions and use them to obtain the reliability index of the components instead of counting the number of failures during the simulation process. This way, it is possible to obtain a proper estimation of the reliability index by considerably smaller number of simulations where it might be even possible to use a commercial FEM package. In general, number of simulations in the range of 1000 to 10000 seems to yield reasonably accurate results.

- Two major methods can be employed when a first order reliability analysis method is intended. The difference between the methods is in the way the derivatives of the performance function are computed. Using distinctive sensitivity analysis methods leads to different methodologies. If a finite difference method is utilised, there is no need for alterations to the finite element code. It also makes it possible to use a commercial finite element package. If a classical perturbation method is used, alterations to the finite element code are necessary. In case the former method together with a commercial finite element package is used, the efficiency and speed of calculation can be hundreds of time less than the case where latter method is used. For an integrated reliability analysis computer environment the latter method is recommended.

- For component level reliability analysis of the compression members, the value of χ that further decreases the resistance of compression members should be treated as a random variable. Although considering χ as a deterministic variable makes the calculations and programming more straightforward, especially if a system reliability analysis is also intended, it is shown that assuming the value of χ as a deterministic value might lead to an overestimation of RI values which in some cases is significant.

- For the development of an integrated stochastic analysis of the truss structure, a method such as FSFEM can be used for a preliminary estimation of component reliability indices since it is fast and efficient. A simulation method should also be provided for the verification of the results obtained through FSFEM.

8.2.2 System level reliability evaluation

The recommendations with regard to the reliability analysis are given below.

- For system reliability analysis the β-unzipping method seems to provide a more convenient algorithmic approach compared to a method such as the branch and bound search method. The β-unzipping method follows a "smooth" linear procedure for system reliability analysis. Different levels of system reliability analysis are investigated in an orderly manner. In the branch and bound search method, however, the search doesn't follow a "smooth" procedure; shifts between different damaged states at different levels are often necessary for the identification of the failure paths. In general, an automatic system reliability analysis is easier if the β-unzipping approach is selected.

- For the system reliability analysis, it is suggested that possible mechanisms be identified before the analysis of the system. Some sequences of failure may not be consistent with the expected degree of redundancy of the structure. Thus, it is recommended to introduce a step before each level prompting the user to define certain failure sequences to be ignored for a certain level of system reliability analysis.

- The branch and bound search method is more robust in identification of the failure paths compared to the β-unzipping method. The β-unzipping method is highly dependent on the definition of the levels. Failure mechanisms can form at an earlier level, but due to the structural degree of redundancy the reliability evaluation can further to the next level. Nevertheless, the branch and bound method defines a failure mechanism as the attainment of a terminal node regardless of the failure level of the node in the structural failure tree.

- In the process of system reliability computations in damaged states of the structure, some failure modes may not be applicable. In a computerised application of the methodology these inapplicable failure modes mostly cause the divergence of the FORM algorithm. Certain error codes can be considered in the computerised analysis to ignore these certain failure modes.

- A good approach for the identification of unimportant or inapplicable limit state functions for certain damaged states of the structure is to assume that all the basic random variables are normally distributed. This way, the limit state functions of the members can be evaluated using a FOSM formulation where the iterative procedure of FORM method can be skipped. This helps identify members that give unreasonable reliability index values for certain failure modes where unreasonable RI values mean that the reliability index can be either too high or even negative. These cases usually cause the divergence of the FORM method if the basic input random variables are not normally distributed.

- Care should be taken with regards to the different failure sequences that lead to the same physical damaged state of the structure. It is of paramount importance to take note of the fact that these failure sequences should be considered separately and shouldn't be ignored simply because they refer to the same physical damaged state. It is observed that the failure sequences that are composed of the same failure elements can yield different reliability indices for a certain damaged state. This is due to the fact that the damaged state is reached through different sequences of element failures.

- For a computerised system reliability analysis four major modules should be provided. A general intact structure module can be used as a central module where the structural system as well as the basic input random variables can be defined. The initial Finite Element Analysis (FEA) of the intact structure can also be performed in this module. A damaged structure module can be used for developing a FEM model for different damaged states of the structure. The damaged states can be analysed in this module. A component level reliability analysis module is needed to perform the reliability evaluation of the limit state functions of the members. This module has to be linked to the general module and the damaged state module. The results of the FEM analyses regarding the member influence factors are then received for the reliability analysis. A system reliability module is also required. This module can be used for the generation of the equivalent safety margins of the parallel systems or element failure sequences as well as the system reliability calculation for the parallel systems.

8.3 Suggestions for further research and study

The structural reliability is a very broad field of study. Regarding a computerised and algorithmic stochastic reliability analysis of structures the following proposals are made:

- In practice, for truss structures, usually the chord members are modelled as continuous members where they might act as flexural members rather than members that are purely under axial tractions. For a computerised component and system level stochastic analysis of trusses, the methodologies can be modified to accommodate the combination of flexural and axial actions for the chord members. This includes modification to the finite element codes as well as the limit state functions of these members where the resistances should be modified. For the system reliability analysis, the plastic analysis (for the chord members) should include the insertion of plastic hinges as well as axial hinges based on the failure mode of the members.

- The methodologies developed and investigated throughout this study should be extended to other structures such as bridges and moment-resisting frames. The component level reliability analysis methodologies can be extended to other types of structures with modification to the finite element analysis and component resistances. System level reliability analysis methods such as the β-unzipping method can also be extend to investigate the system reliability of the moment-resisting frame or super structure of bridges (such as a bridge superstructure composed of a deck supported by girders).

- In general, an integrated stochastic finite element analysis environment should be developed where a component and system level reliability analysis of structures could be performed. Such an environment makes it possible to do a more realistic structural safety analysis where the uncertainties regarding loads and resistances are taken into account. This way, it is also possible to perform an inspection, repair, and maintenance optimisation for structures such as bridges based on the stochastic evaluation of the structure.

- The values of $\Delta\beta$ were chosen arbitrarily in the process of system reliability calculation using the β-unzipping method. However, a sensitivity analysis can be performed on the effect of $\Delta\beta$ values on the system reliability bounds as well as the system failure tree.

Bibliography

[1] *Documentation for the Strand7 Application Programming Interface*. Sydney Austrailia, 2010.

[2] P.R. Adduri. *Robust estimation of reliability in the presence of multiple failure modes*. PhD thesis, Wright State University, 2006.

[3] H.L. Anderson. Metropolis Monte Carlo and the MANIAC. *Los Alamos Science*, pages 96–108, 1986.

[4] Anon. *SANS 10162-1:2005. Part 1: Limit state design of hot-rolled Steel work*. Standards South Africa, 2005.

[5] Richard Baker and Jay Puckett. *Bridge management*. John Wiley & Sons, 2nd edition edition, 2001.

[6] Y.F. Bejerger. Application of system reliability analysis to offshore structures. Technical Report 7G, Stanford University, California, U.S.A, 1984.

[7] P. Bjerager. Reliability analysis of structural systems. Technical report, Department of structural engineering , Technical university of Denmark, 1984.

[8] R.D. Cook, D.S. Malkus, M.E. Plesha, and R.J. Witt. *Concepts and Applications of Finite Element Analysis*. John Wiley & sons, 2002.

[9] C. A. Cornell. A probability-based structural code. *ACI Journal*, pages 974–985, 1969.

[10] O. Ditlevsen. Narrow reliability bounds for structural systems. *Structural Mechanics*, 7(4):453–472, 1979.

[11] C.W.L Dunnett and M. Sobel. Approximations to the probability of integeral and certain percentage point of a multivariate analogue of students's t-distribution. *Bitometrika*, 42:258–260, 1979.

[12] A.C. Estes and D.M. Frangopol. Using system reliability to evaluate and maintain structural systems. *Computational Structural Engineering*, (1):71–80, 2001.

[13] A.C. Estes and D.M. Frangopol. Life-cycle evaluation and condition assessment of structures. *CRC Press*, 2005.

[14] B. Ellingwood et al. *Development of a probability based load criterion for American national standard A58*. Dept. of Commerce, National Bureau of Standards, Washington, 1980.

[15] A.J.M. Ferreira. *MATLAB Codes for Finite Eelement Analysis*. Springer, 2009.

[16] A. Florian. An efficient sampling scheme: Updated Latin Hypercube sampling. *Probabilistic Engineering Mechanics*, 7(2):123–130, 1992.

[17] T.V. Galambos, B. Ellingwood, J.G MacGregor, and C.A. Cornell. *Development of a probability based load criterion for American national standard A58 : building code requirements for minimum design loads in buildings and other structures*. Dept. of Commerce, National Bureau of Standards, 1980.

[18] S. Gollwitzer and R. Rackwitz. Equivalent components in first-order system reliability. *Reliability Engineering*, 5:99–115, 1983.

[19] A. Haldar and S. Mahadevan. *Reliability Assessment Using Stochastic Finite Element Analysis*. John Wiley & Sons, 2000.

[20] S. Hao. I-35w bridge collapse. *Journal of Bridge Engineering*, 1010.

[21] A.M. Hasofer and N.C. Lind. An exact and invariant first order reliability format. *Proc. ASCE, Journal Engineering Mechanics Div.*, pages 111–121, 1974.

[22] M. Holicky. *Reliability Analysis for Structural Design*. SUN MeDIA, Stellenbosch, 1st edition edition, 2009.

[23] A. Karamchandani. Structural system reliability analysis methods. Technical Report 83, Standford University, 1987.

[24] D.S. Kim, S.Y. OK, J. Song, and H.M. Koh. System reliability analysis using dominant failure modes identified by selective searching technique. *Reliability Engineering and system safety*, 2013.

[25] A. Der Kiureghian and J-B Ke. The stochastic finite element method in structural reliability. *Probabilistic Engineering Mechanics*, 3(2):82–91, 1988.

[26] S. Krenk and J. Hogsberg. *Statics and Mechanics of Structures*, chapter 2. Springer, 2013.

[27] G.Q. Li and J.J. Li. A semi-analytical simulation method for reliability assessement of structural sysems. *Reliability Engineering and system safety*, 2002.

[28] G.Q. Li and J.J. Li. *Advanced Analysis and Design of Steel Frames*. John Wiley & sons, 2007.

[29] R. Loov. A simple equation for axially loaded steel column. *Candian Journal of Civil Engineering*, pages 272–276, 1996.

[30] M.D. McKay, R.J. Beckman, and W.J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

[31] R.E. Melchers and L.K. Tang. Dominant failure modes in stochastic structural systems. *Structural Safety*, pages 127–143, 1984.

[32] Robert E. Melchers. *Structural Reliability Analysis and Prediction*. John Wiley & Sons, Chichester, West Sussex, 2nd edition edition, 1999.

[33] D.C. Montgomery and G.C. Runger. *Applied statistics and probability for engineers*. John Wiley & Sons, 2006.

[34] F. Moses. System reliability developments in structural engineering. *Structural Safety*, pages 3–13, 1982.

[35] F. Moses and B. Stahl. Reliability analysis format for offshore structures. In *Offshore Technology Conference*, Houston, Texas, 1978.

[36] E. Nikoladis, D.M. Ghiocel, and S. Singhal, editors. *Engineering Design Reliability Handbook*. CRC Press, 2004.

[37] S.A. Nowak and K.R. Collins. *Reliability of Structures*. McGraw-Hill, 2000.

[38] R. Rackwitz. Practical probabilistic approach to design. *Comite European du Beton*, (112), 1976.

[39] R. Rackwitz and B. Flessler. Structural reliability under combined random load sequences. *Computers & Structures*, 9(5):489–494, 1978.

[40] R. Rashedi and F. Moses. Identificatio of failure modes in system reliability. *Journal of structural engineering*, 114(2), 1988.

[41] E. Rosenbleuth and L. Esteva. Reliability bases for some mexican codes. *ACI Journal*, pages 1–41, 1972.

[42] M. J. Ryall. *Design of Highway Bridges*. Butterworth-Heinemann, 1st edition edition, 2007.

[43] S. Shao and Y. Murotsu. Approach to failure mode analysis of large structures. *Probablistic engineering mechanics*, 1999.

[44] B. Sudret and A. Der Kiureghian. The stochastic finite element method in structural reliability (a state-of-the-art report). Technical report, University of California, Berkeley, 2000.

[45] P. Thoft-Christensen. The β-unzipping method. Technical report, Institute of Building Technology and Structural Engineering, Alborg University Center, 1982.

[46] P. Thoft-Christensen and M.J. Baker. *Structural Reliability Theory and Its Applications*. Springer-Verlag, 1982.

[47] P. Thoft-Christensen and Y. Murotsu. Application of structural systems reliability theory. *Springer Verlag*, 1986.

[48] D. Wei and S. Rahman. Structural reliability analysis by univariate decomposition and numerical integration. *Prbabilistic Engineering Mechancis*, pages 27–38, 2006.

[49] F.K. Wittle. Seminar on the finite element method and analysis of systems with uncertain properties. ETH Zurich, 2007.

[50] M.B. Wong. *Plastic Analysis and Design of Steel Structures*. Butterworth-Heinemann, 2009.

[51] Q. Xiao and S. Mahadevan. Fast failure mode identification for ductile structural system reliability. *Structural Safety*, 1994.

[52] Du Xiaoping. *Probablistic Engineering Design*, chapter 7. Univeristy of Missori-Rolla, 2005.

Appendices

Appendix I

I.1 Strand 7 Application Programming Interface (API)

For the component level reliability analysis which uses the finite difference method as the main method of implementing a stochastic finite element, one solution was to link the reliability evaluation programming code to the finite element model of the structure (FD-SFEM). One of the available FEM software packages at the University of Stellenobsch is the Srand7 version 2.4.2. It was figured that Strand7 provides a very effective application programming interface which is relatively easy to utilise, although it has its own complication. Below is a brief explanation concerning how to use the Strand7 API for the post-processing of FEM analysis results with regards to a component level reliability evaluation. Some of the material is taken from Strand7 API user manual together with Matlab's product help, and some is the author's experience.

Using the Strand7 API, it is possible to link the Matlab program developed for the FORM reliability analysis which uses the finite difference sensitivity approach(FD-SFEM) to the finite element model of the structure in order to access the finite element analysis data, such as component internal forces. Strand7 contains a dynamic link library file (DLL file) which is called "St7API.dll". There are also a number of header files ("filename.h") and constant files provided by the developer of the software. The aforementioned dynamic link library file provides the capability to: read strand7 finite element data; modify or create Strand7 finite element data; launch the Strand7 solvers; and read Strand7 result data [1]. The intermediate between the programming language and the "ST7API.dll" file is the header file. The header file defines all the constants used and the function calling conventions for each supported language. Strand7 version 2.4.2 includes a header file for Matlab so that it will be possible to use Matlab together with Strand7.

The following steps are suggested in order to be able to set up a program for using the Strand7 API.

1. The finite element program has to be properly licensed for the API to be used. Fortunately, the license was already provided by the University of Stellenbosch, and through connection to the university network one can effectively use the FEM software.

2. A very important issue in using strand7 API is for the program to be in a directory within

the windows search path. In order to introduce the St7API.dll file to the search path of the windows, the following steps should be performed:

(a) Go to control panel\system \advance system settings \environmental variables \system variables.

(b) Scroll down to find “Path”.

(c) Click on Edit.

(d) In the opened window add the directory shown inside the quotation marks “;c:\\program files\Strand7 R24\bin”.

(e) click “OK” and restart the computer.

Warning: be careful not to erase the existing directories while editing the search path of the windows. Just add the directory that is mentioned above right to the end of the existing directories.

3. To use the API the Header file is indispensable. It declares all the exported function calls in St7API.dll. For Matlab the Header file is shown as “St7APICall.h”.

4. For some languages the windows API call “LOADLIBRARY” is needed. That is not the case if Matlab is used.

5. Another important file is the constants file. It is about the pre-defined constants for the functions in the API. For Matlab “St7APIConst.m” file is provided. It is merely a Matlab m-file containing all the constants of the API functions.

I.2 Connecting Strand7 API to Matlab

In order to connect Matlab to Strand7 the two Header and Constant file are necessary. These files are named as “St7ApiConst.m” and “St7APICall.h”. The two files can be found in the directory “c:\Program Files (x86)\Strand7 R24\API\Includes\Matlab”. It is recommended that this directory be add to Matlab’s search path.

For the purpose of loading the Strand7 API in Matlab some built-in functions are provided. One of these built-in functions is “loadlibrary”. It follows the format “loadlibrary(‘shrlib’,‘hfile’)”; by using this built-in function, the functions in the Header file (“hfile”) that are sheared in the library (“.dll file”) will be loaded. Sometimes a compiling error may emerge or Matlab may ask for a C compiler. However, a C compiler can be easily set up via “mex-setup” function of Matlab. Another essential Matlab built-in function is “libisloaded” function. This function is a logical function, and it checks if the library is loaded or not. If the library is loaded it returns 1 and if not it returns 0. Clearly in order to unload the library the function “unloadlibrary” can be used. Here only the library name is necessary in order to unload the library, so it has the form “unloadlibrary (‘libname’)”. Eventually, one of the most useful Matlab built-in functions, which is essential to use the API, is the “calllib” function. In order to call the functions provided in the Strand7 API this built-in Matlab function must be utilised. It has the form “Calllib(‘libname’,‘function’,‘arg1,...,argN’)”.

For further information regarding these built-in Matlab function, the reader is referred to Matlab product help.

I.3 Procedure of Finite Element Programming For a Truss Structure Using the Strand 7 API

The required steps for doing a finite element analysis for a truss structure will be outlined below. This process is applicable to other types of structures as well, only different types of functions have to be called from the Strand7 API library. These steps are as below:

1. The first step is to build the finite element model in the graphical user interface of the Strand7 software. A step-wise procedure building a truss model is given beneath:

   (a) Defining units.

   (b) Creating nodes (nodes can also be replicated).

   (c) Creating beam elements.

   (d) Setting freedom cases.

   (e) Assigning boundary conditions.

   (f) Assigning loads.

   (g) Property input (selecting the geometry and type of the beams).

   (h) * solving the finite element model (linear static solver).

   (i) * post processing the finite element model (opening the result file, applying load combination and so on).

   The steps that are shown above with asterisks will be performed through the Strand7 application programming interface. All the steps mentioned above can be altered by functions and routines specified inside the Strand7 API.

2. Next, the Strand7 API has to be loaded. This is done using the proper built-in Matlab functions that were mentioned in the previous section. Also a folder should be created for temporary storage.

3. In this step, firstly, the finite element file which is a file with the extension ".st7" has to be opened. This task can be done using the API routine "ST7OpenFile".

4. The linear static solver should be called in order to perform the finite element analysis of the model. The following API routines are useful for this task: "St7SetResultFileName"; "St7SetSolverScheme"; "St7setSolverSort"; "St7SetEntityResult"; "St7RunSolver". The first four routines are utilised to perform the essential settings, and the last routine is used to run the solver. All of these settings are also available in the graphical user interface of the software.

5. Getting access to the results: For this purpose the function "St7OpenResultFile" is used to open the results file model. After Strand7 finishes the analysis, it creates a results file with the extension ".lsa". With the API routine mentioned above, it is possible to open this file and get access to its content. Nevertheless, calling any specific result type out of the results file necessitates the inclusion of other API routines mentioned in next step.

6. Extracting the desired results out of the results file: After the results file is opened, specific types of results will be available. For instance, in order to get the internal force of the truss members the API subroutine "St7GetBeamResultsArray" is utilised. This API routine gives the internal forces in the form of an array of data.

About the whole process of programming with Matlab using the Strand7 API some notes are necessary to be mentioned:

- If the programming is done by defining different functions, it might be required to call the "St7APIConst" file in some functions.

- All the API routines that are needed to be used inside a function have to be declared "Global" so that Matlab can recognise them and any further errors are avoided.

- It is good practice to define a function that catches the possible errors inside the API and converts them to Matlab errors. This can be done by using "St7GetAPIErrorString" and "St7GetSolverErrorString" API routines.

The whole process can be depicted as a flowchart as shown in the figure below. It should also be mentioned that the flowchart for the SFEM method is shown in Chapter 3.

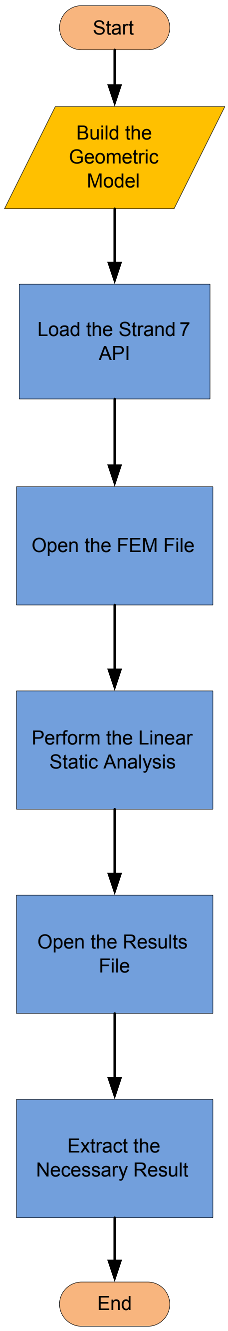

Figure 1: Flowchart of FEM analysis post-processing using Strand7 API

Appendix II

Here are the different Matlab transcripts that were used for the component level reliability evaluation. It includes all the "m-files" that were created for the reliability calculations presented in Chapter 5.

II.4 Matlab transcript for the FEM analysis using Strand7 version 2.4.2 API

This section shows the programs that are used for the finite element analysis using Strand7 version 2.4.2 application programming interface. Two "m-files" were created for this purpose: one was created for the 3-bar statically determinate truss structure, and the other for the 10-bar statically indeterminate structure.

II.4.1 Main Program

The following are the main transcripts for the deterministic finite element analysis.

II.4.1.1 3-bar statically determinate truss structure

The "m-file" for the finite element analysis of the 3-bar truss structure shown in Chapter 5.

```
%function file that performs the FEM analysis of a simple truss
%Using the Strand7 version 2.4.2 API to apply a Finite difference Stochastic ...
    finite
%element method to calculate the reliability of a simple truss

function [F,nBeams] = FEManalysis(force1)

global uID %Identification of the file


ScratchPath = 'C:\Temp';
```

```
% Load the St7API constants as MATLAB global variables
uID = 1;
St7APIConst();

% Load the api if not already loaded if not then load the header file as
% well
fprintf('Loading ST7API.DLL... ')
if ~libisloaded('St7API')
    loadlibrary('St7API.dll', 'St7APICall.h');
    iErr = calllib('St7API', 'St7Init');
    HandleError1(iErr);
end

% Wrap everything in try—catch so that the St7API can be correctly unloaded
% in the event of a MATLAB error
try

% opening the file and running the solver

global stLinearStaticSolver smProgressRun stSparse rnAMD srBeamForce btFalse
uID = 1;
%%%%%%%%%%%%%%%%%%%%%%%%%1.OPENING THE MODEL%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Open ST7 model file
bal = ['\simple truss.lsa'];
FileName = [pwd, '\simple truss.st7'];
ResultName = [pwd, bal];
iErr = calllib('St7API', 'St7OpenFile', uID, FileName, ScratchPath);
HandleError1(iErr)
iErr = calllib('St7API','St7SetNodeForce3',uID,2,1,[0,force1,0]);
HandleError1(iErr);
%%%%%%%%%%%%%%%%2.CALLING THE SOLVER TO SOLVE THE MODEL%%%%%%%%%%%%%%%%
% Call the solver
fprintf('Running the Linear Static Solver... ');
iErr = calllib('St7API', 'St7SetResultFileName', uID, ResultName);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetSolverScheme', uID, stSparse);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetSolverSort', uID, rnAMD);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetEntityResult', uID, srBeamForce, 1);
HandleError1(iErr);
iErr = calllib('St7API', 'St7RunSolver', uID, stLinearStaticSolver, smProgressRun...
    , 1);
HandleError1(iErr);
fprintf('Done \n')

%%%%%%%%%%%%%%%%%%%%%%%%%%3.OPENING THE RESULT FILE%%%%%%%%%%%%%%%%%%%%%
fprintf('Processing Results... ');
```

```
% St001 : Change boolean to btFALSE
numPrimary = 0;
numSecondary = 0;
[iErr, numPrimary, numSecondary] = calllib('St7API', 'St7OpenResultFile', uID, ...
    ResultName, '', btFalse, 0, 0);
HandleError1(iErr);

%%%%%%%%%%%%%%%%%%%%%%%%4.EXTRACTING THE REQUIRED REULTS%%%%%%%%%%%%%%%
%extracting the forces acting on the members due to tha applied load out of
%the results file

global tyBEAM rtBeamForce stBeamLocal ipAxial kMaxBeamResult

Beamres = zeros(kMaxBeamResult);

% Get number of Beams
fprintf('    Counting beams... \n');
[iErr, nBeams] = calllib('St7API', 'St7GetTotal', uID, tyBEAM, 0);
HandleError1(iErr);


% catch case where no beams are found
if (nBeams == 0)
    F = 0;
    fprintf('    No beams. Returning. \n');
    return;
end

% Changed stations to the default of 5
nstations = 5;

% beampos must be an array
Beampos =  zeros(1,nstations);
ncolumns = 6;

% beamres must be an array
Beamres = zeros(1,kMaxBeamResult);
fprintf('    Getting results for %i beams...\n',nBeams);
for i = 1:nBeams
    fprintf('        Getting results beam %i...\n',i);
    % St001 : Changed scalar output parameters to 0 in the input param list
    [iErr,nstations,ncolumns, Beampos, Beamres] = calllib('St7API', '...
       St7GetBeamResultArray', uID, rtBeamForce, stBeamLocal, i, 2, ...
                                                                      1,0,0,...
                                                                         Beampos...
                                                                         ,...
                                                                         Beamres...
                                                                         );
    HandleError1(iErr);
```

```
    F(i)=Beamres(ipAxial+1);

end     % end for

CloseAndUnload1()

catch

   CloseAndUnload1()
   rethrow(lasterror)

end
```

II.4.1.2 10-bar statically indeterminate truss structure

The “m-file” for the finite element analysis of the 10-bar truss structure shown in Chapter 5.

```
%function file that Performs the FEM analysis for the ten bar truss
%Using the Strand7 API to apply a Finite difference Stochastic finite
%element method to calculate the reliability of the truss components

% force1 is a row vector with two components for applying the force on
% nodes 2 and 3 of the ten-bar truss

function [F,nBeams] = tenbartruss(force1)

global uID %Identification of the file


ScratchPath = 'C:\Temp';


% Load the St7API constants as MATLAB global variables
uID = 1;
St7APIConst();

% Load the api if not already loaded if not then load the header file as
% well
fprintf('Loading ST7API.DLL... ')
if ~libisloaded('St7API')
    loadlibrary('St7API.dll', 'St7APICall.h');
    iErr = calllib('St7API', 'St7Init');
    HandleError1(iErr);
end

% Wrap everything in try-catch so that the St7API can be correctly unloaded
% in the event of a MATLAB error
try
```

```
% opening the file and running the solver

global stLinearStaticSolver smBackgroundRun  stSparse rnAMD srBeamForce btFalse
uID = 1;
%%%%%%%%%%%%%%%%%%%%%%%%%1.OPENING THE MODEL%%%%%%%%%%%%%%%%%%%%%%%%%%
% Open ST7 model file
bal = ['\10bartruss-m.lsa'];
FileName = [pwd, '\10bartruss-m.st7'];
ResultName = [pwd, bal];
iErr = calllib('St7API', 'St7OpenFile', uID, FileName, ScratchPath);
HandleError1(iErr)
iErr = calllib('St7API','St7SetNodeForce3',uID,2,1,[0,force1(1),0]);
HandleError1(iErr);
iErr = calllib('St7API','St7SetNodeForce3',uID,3,1,[0,force1(2),0]);
HandleError1(iErr);
%%%%%%%%%%%%%%%2.CALLING THE SOLVER TO SOLVE THE MODEL%%%%%%%%%%%%%%%
% Call the solver
fprintf('Running the Linear Static Solver... ');
iErr = calllib('St7API', 'St7SetResultFileName', uID, ResultName);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetSolverScheme', uID, stSparse);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetSolverSort', uID, rnAMD);
HandleError1(iErr);
iErr = calllib('St7API', 'St7SetEntityResult', uID, srBeamForce, 1);
HandleError1(iErr);
iErr = calllib('St7API', 'St7RunSolver', uID, stLinearStaticSolver, ...
    smBackgroundRun, 1);
HandleError1(iErr);
fprintf('Done \n')

%%%%%%%%%%%%%%%%%%%%%%%%%%3.OPENING THE RESULT FILE%%%%%%%%%%%%%%%%%%%%%
fprintf('Processing Results... ');

% St001 : Change boolean to btFALSE
numPrimary = 0;
numSecondary = 0;
[iErr, numPrimary, numSecondary] = calllib('St7API', 'St7OpenResultFile', uID, ...
    ResultName, '', btFalse, 0, 0);
HandleError1(iErr);

%%%%%%%%%%%%%%%%%%%%%%%%4.EXTRACTING THE REQUIRED REULTS%%%%%%%%%%%%%%%
%extracting the forces acting on the members due to tha applied load out of
%the results file

global tyBEAM rtBeamForce stBeamLocal ipAxial kMaxBeamResult

Beamres = zeros(kMaxBeamResult);
```

```
% Get number of Beams
fprintf('    Counting beams... \n');
[iErr, nBeams] = calllib('St7API', 'St7GetTotal', uID, tyBEAM, 0);
HandleError1(iErr);


% catch case where no beams are found
if (nBeams == 0)
    F = 0;
    fprintf('    No beams. Returning. \n');
    return;
end

% Changed stations to the default of 5
nstations = 5;

% beampos must be an array
Beampos =  zeros(1,nstations);
ncolumns = 6;

% beamres must be an array
Beamres = zeros(1,kMaxBeamResult);
fprintf('    Getting results for %i beams...\n',nBeams);
for i = 1:nBeams
    fprintf('        Getting results beam %i...\n',i);
    % St001 : Changed scalar output parameters to 0 in the input param list
    [iErr,nstations,ncolumns, Beampos, Beamres] = calllib('St7API', '...
        St7GetBeamResultArray', uID, rtBeamForce, stBeamLocal, i, 2, ...
                                                                        1,0,0,...
                                                                            Beampos...
                                                                            ,...
                                                                            Beamres...
                                                                            );
    HandleError1(iErr);
    F(i)=Beamres(ipAxial+1);

end     % end for

CloseAndUnload1()

catch

   CloseAndUnload1()
   rethrow(lasterror)

end
```

II.4.2 Function to Handel Errors in API

This function is defined to convert the API errors into Matlab codes.

```
function HandleError1(iErr)
% Helper to convert ST7API error codes to MATLAB errors

global kMaxStrLen ERR7_NoError

if (iErr~=ERR7_NoError)
    str = char(zeros(kMaxStrLen, 1));
    [iNewErr, str] = calllib('St7API', 'St7GetAPIErrorString', iErr, str, ...
        kMaxStrLen);
    if (iNewErr>0)
        [iNewErr, str] = calllib('St7API', 'ST7GetSolverErrorString', iErr ,str, ...
            kMaxStrLen);
    end
    % Issue as a MATLAB error
    error(['St7API error: ', str]);
end

end      % HandleError()
```

II.4.3 Function to Close and unload

This function was defined to close all the files with respect to the assigned "uId", and to unload the library as well ("uId" is an arbitrarily selected number assigned to the finite element model).

```
function CloseAndUnload1()
% Close any open files associated with uID and unload the St7API.

global uID

calllib('St7API', 'St7CloseResultFile', uID);
calllib('St7API','St7CloseFile',uID);
unloadlibrary('St7API');

end      % CloseAndUnload()
```

II.4.4 Function for section properties

This function reads the section properties of the finite element model. Properties such as element areas, Young's modulus,... are taken from the defined model.

```
%function file to extract the model properties of the structure defined in
%Strand7

function [L,BeamData,Modulus,nBeams] = secprop()

global uID %Identification of the file


ScratchPath = 'C:\Temp';


% Load the St7API constants as MATLAB global variables
uID = 1;
St7APIConst();

% Load the api if not already loaded if not then load the header file as
% well
fprintf('Loading ST7API.DLL... ')
if ~libisloaded('St7API')
    loadlibrary('St7API.dll', 'St7APICall.h');
    iErr = calllib('St7API', 'St7Init');
    HandleError1(iErr);
end

% Wrap everything in try-catch so that the St7API can be correctly unloaded
% in the event of a MATLAB error
try

% opening the file and running the solver

uID = 1;
%%%%%%%%%%%%%%%%%%%%%%%%1.OPENING THE MODEL%%%%%%%%%%%%%%%%%%%%%%%%%
% Open ST7 model file
FileName = [pwd, '\10bartruss-m.st7'];
iErr = calllib('St7API', 'St7OpenFile', uID, FileName, ScratchPath);
HandleError1(iErr)

%%%%%%%%%%%%%%%%2.CALLING THE SOLVER TO SOLVE THE MODEL%%%%%%%%%%%%%%%

global tyBEAM

nBeams = 0;
disp('    Counting beams...');
[iErr, nBeams] = calllib('St7API', 'St7GetTotal', uID, tyBEAM, 0);
HandleError1(iErr);
L = zeros(1,nBeams);
leg1 = 0;
for i = 1:nBeams

    [iErr, leg1] = calllib('St7API', 'St7GetElementData', uID, tyBEAM, i,leg1);
```

```
    L(i) = leg1;
    HandleError1(iErr);

end

beamD = zeros(1,3);
secD = zeros(1,19);
matD = zeros(1,3);
BeamData = zeros(nBeams,3);

for i = 1:nBeams

   [iErr,beamD,secD, matD] = calllib('St7API', 'St7GetBeamPropertyData', uID, i, ...
       beamD,secD,matD);
   HandleError1(iErr);
   BeamData(i,:)= secD([1,2,3]);
   Modulus = matD(1);
end

CloseAndUnload1()

catch

   CloseAndUnload1()
   rethrow(lasterror)

end
```

II.5 Matlab transcripts for the finite element analysis of the 10-bar truss structure

A program was written in Matlab for the finite element analysis of the 10-bar truss structure [15]. This program has one main "m-file" and 3 other functions:

- "FEMcodetruss" which is the main program where the model is defined and the three other functions are connected.
- "formstiffness2Dtruss" which is the program that assembles the structural stiffness matrix.
- "solution" which calculates the displacements based on the structural stiffness matrix and the applied loads matrix.
- "stresses2Dtruss" which is the function where the response of the structure is formed i.e. element stresses.

These transcripts are present below:

II.5.1 Main program

```
%......................................................................

%Matlab Code for analysis of the ten-bar truss

% E : Modulous of Elasticity
% A : Areas of the members
% L : Length of the bars
% E = input('Enter Modulus of Ealsticity: ');
%
% A = input('Enter the vector of member areas: ');
function mforce = FEMcodetruss(F)

E = 2*10^11;

A = [0.001961,0.000929,0.000188,0.000188,0.001226,0.00227,0.001049,0.000331,...
    0.000929,0.000778];
% Creating the geometrical Model

numberElements = 10;
numberNodes = 6;

elementNodes = [1 2;2 3;3 6;5 6;4 5;1 5;4 2;2 5;2 6;5 3];
nodeCoordinates = [0 0;3 0;6 0;0 4;3 4;6 4];

xx = nodeCoordinates(:,1);
yy = nodeCoordinates(:,2);

% Forming the displacement vector, load vector and stiffness matrix

GDof = 2*numberNodes;
displacements = zeros(GDof,1);
force = zeros(GDof,1);

% applying the load on the structure

force(4)  = F(1)*1000;
force(6)  = F(2)*1000;

[stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A);

% Difnining boundry conditons

prescribedDof = [1 2 7 8];

% calculating the displacements stresses and intenal forces
```

```
displacements = solution(GDof,prescribedDof,stiffness,force);

[sigma,mforce] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A);

 mforce = mforce/1000;
end
```

II.5.2 Structural stiffness formation function

```
% funciton to assembel the system stiffness matrix

function [stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A)

stiffness = zeros(GDof);
%calculating the system stiffness matrix

for e = 1:numberElements;

    indice = elementNodes(e,:);

    %figuring out element dgrees of freedom

    elementDof = [indice(1)*2-1 indice(1)*2 indice(2)*2-1 indice(2)*2];

    xa = xx(indice(2))- xx(indice(1));
    ya = yy(indice(2))- yy(indice(1));
    length_element = sqrt(xa*xa + ya*ya);

    C = xa/length_element;
    S = ya/length_element;

    k1= (E*A(e))/length_element * [C*C C*S -C*C -C*S; C*S S*S -C*S -S*S;...
        -C*C -C*S C*C C*S;-C*S -S*S C*S S*S];

    stiffness(elementDof,elementDof) = stiffness(elementDof,elementDof) + k1;

end
```

II.5.3 Displacements computation function

```
% Function to get the solution in the global dgerees of freedom

function displacement = solution(GDof,prescribedDof,stiffness,force)
```

```
activeDof = setdiff(1:GDof,prescribedDof);

U = stiffness(activeDof,activeDof)\(force(activeDof));

displacement = zeros(GDof,1);
displacement(activeDof) = U;
```

II.5.4 Structural response calculation function

```
%Calculating the stresses and forces at elements

function [sigma,mforce] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A)

for e = 1 :numberElements

   indice = elementNodes(e,:);
   % getting the degreess of freedom of the elments
   elementDof = [indice(1)*2-1 indice(1)*2 indice(2)*2-1 indice(2)*2];
   % getting the length of the element
   xa = xx(indice(2))-xx(indice(1));
   ya = yy(indice(2)) - yy(indice(1));

   length_element = sqrt(xa*xa + ya*ya);

   C = xa/length_element;
   S = ya/length_element;

   sigma(e) = (E/length_element)*[-C -S C S]*displacements(elementDof);

   mforce(e) = sigma(e)* A(e);
end
```

II.6 Transcripts for the component reliability analysis

Two structures were considered in Chapter 5 for the component reliability analysis: a 3-bar statically determinate truss structure and a 10-bar statically indeterminate truss structure. Each of these structures were analysed using different reliability analysis methods. The Matlab programming transcripts are given below.

II.6.1 Transcripts for the reliability analysis of the 3-bar truss structure

This structure was analysed using 3 different methods:

- The FORM analysis based on finite difference sensitivity analysis.

- Direct Monte Carlo simulation.
- Integration method.

The transcripts for these methods are given respectively.

II.6.1.1 Matlab transcript for the FORM reliability analysis

```
% applying the FORM analysis for a simple truss structure
% Using the strand7 API
% a function is defined in matlab which uses the strand 7 API to do a FEM
% analysis of the truss model

%%%%%%%%%%%%%%%%%%%%%%Start of the FORM analysis%%%%%%%%%%%%%%%%%%%%%%
%entering the value of the applied load

F = input('Enter The mean value of the load applied on the truss: ');

%entering the value of the standard deviation

sigma = input('Enter the Standard Deviation of the applied load: ');

[mforce,n] = FEManalysis(F);
f = F;
%Creating place holder for the variables
b = [0,0,0];
g1 = zeros(1,n);
muN = zeros(1,n);
sigmaN = zeros(1,n);
p = zeros(1,n);
g2 = zeros(1,n);
delG = zeros(1,n);
delGN = zeros(1,n);
Pnew = zeros(1,n);
beta = zeros(1,n);
beta1 = zeros(1,n);
g3 = zeros(1,n);
%counting the time it takes to do the reliability analysis
tic;
for i = 1:n
    F = f;
    [mforce,n] = FEManalysis(F);
    R1 = 170.12; %deterministic compression resistance(buckling)
    R2 = 130; %deterministic tension resistance
    beta1(i) = 0;
    resBeta = 20;
    g3(i) = 100;
```

```
%forming a while loop to check convergence criteria
mforce1 = mforce;
while abs(resBeta) > 0.001 && abs(g3(i)) > 0.001
    %check if the member are in tension or compression
    if mforce1(i)< 0

        g1(i) = R2−abs(mforce1(i));
    else
        g1(i) = R1−abs(mforce1(i));
    end

    %calculating the equivalent normal parameters of the variables

    [muN(i),sigmaN(i)] = lognrmeq(F,f,sigma);

    p(i) = (F−muN(i))/sigmaN(i);

    %calculating the partial derivative of the limit state fucntion by
    %reanalyzing the model each time (finite difference mthod)

    percent = 0.01;

    F2 = F + (percent*F);

    [mforce2,nbeam] = FEManalysis(F2);

    if mforce2(i) < 0

        g2(i) = R2−abs(mforce2(i));
    else

        g2(i) = R1−abs(mforce2(i));
    end

    delG(i) = (g2(i)−g1(i))/(abs(mforce2(i)) − abs(mforce1(i)));


    %calculating the partial derivative in the equivalent normal space

    delGN(i) = delG(i)*sigmaN(i);

    %calculating the coordinates of the new design point

    Pnew(i) = (1/(delGN(i)^2))*(delGN(i)*p(i) − g1(i))*delGN(i);

    %calculating the beta value of the member

    beta(i) = sqrt(Pnew(i)^2);

    res_beta = beta(i)−beta1(i);
```

```
        beta1(i)= beta(i);

        %calculating the coordinates of the new iteration point

        F = muN(i)+ sigmaN(i)*Pnew(i);

        [mforce1,n] = FEManalysis(F);
        if mforce1(i)<0

            g3(i) = R2-abs(mforce1(i));
        else
            g3(i) = R1-abs(mforce1(i));
        end
        b(i) = b(i) + 1;
        if b(i) > 20
            disp('The number of iterations exceeds 15');
            break

        end
    end

end
t = toc;
mint = floor(t/60);
sec = floor(t - mint*60);
disp('The Beta values obtained for members are: ');
disp(' ');
disp(beta);
disp(' ')
disp(['Total Number of iteration for FORM analysis: ',num2str(sum(b))]);
disp(' ');
disp(['Elapsed time is: ',num2str(mint),' mins and ',num2str(sec),' secs']);
```

II.6.1.2 Matlab transcript for DMCS

```
%evaluating the reliability of a member of that simple truss using
%Monte carlo simiulation technic

%%%%%%%%%%%%%%%%%%%%%%%%%%Start of the simulation%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
clc
%enter the value of force and standard deviation
mu = input('Enter the value of the applied force: ')

type = input('choose the pdf of F,Normal = 1 Lognormal = 2: ');

sigma = input('Enter the value of the SD: ')
```

```
% n number of random number between 0 and 1

n = input('Enter the number of simulations: ')

% generating pseudo random numbers
a = rand(1,n);
%calculating the inverse of the standard normal variable
z = norminv(a,0,1);

if type == 2
% calculating the lognormal mean and standard deviation

    sigmaln =sqrt(log(1+((sigma^2)/(mu^2))));

    muln = log(mu)-0.5*sigmaln^2;
%generating the random lognormal values for the applied force
    b = muln + sigmaln.*z;
    f = exp(b);
else

    f = z*sigma + mu;
end
%calculating the member force for each lognoraml random variable

mforce = f./(2*sin(0.9273));

g = 170.12 - mforce;

c = g < 0;
nf = sum(c)

pf = nf/n

beta = abs(norminv(pf,0,1))
```

II.6.1.3 Matlab transcript for the integration method

```
%numerical integration of the one of the members in compression
% assuming that the mean value is 170
% sigma is 170x0.2=34
clear all
clc
mu = input('Enter the applied force: ');

sigma = input('Enter the SD value: ');

type = input('Enter the force pdf type normal =1 lognormal=2: ');
```

```
n = input('Enter the numerical integration increment: ');

x = 272:n:3*mu;
if type == 2
   sigmaln =sqrt(log(1+((sigma^2)/(mu^2))));

   muln = log(mu)-0.5*sigmaln^2;

   f = (1/sqrt(2*pi)).*(1./(sigmaln.*x)).*exp(-0.5.*((log(x)-muln)./sigmaln).^2);
else

   f = normpdf(x,mu,sigma);

end
% the dots are added because a scalar is multiplied by a vector dot helps
% matlab to recognize between a matrix product and vector product
Pf = n*sum(f)
beta = abs(norminv(Pf,0,1))
```

II.6.2 Transcripts for the reliability analysis of the 10-bar truss structure

The reliability index values of the components of the 10-bar truss structure were determined using three different methods:

- The FORM reliability analysis using two different sensitivity methods:
  - Finite difference based FORM reliability analysis (FD-SFEM).
  - Classical perturbation based FORM reliability analysis or fully stochastic finite element method (FSFEM).
- Response Surface Method (RSM):
  - RSM using direct Monte Carlo simulation (RSM-DMCS).
  - RSM using Latin Hypercube Sampling Monte Carlo (RSM-LHCSMC).
- Monte Carlo simulation of the whole structure:
  - Direct Monte Carlo Simulation (DMCS).
  - Latin Hypercube Sampling Monte Carlo (LHCSMC).
  - Updated Latin Hypercube Sampling Monte Carlo (ULHCSMC).

The Matlab transcripts for these method are given below.

II.6.2.1 Matlab transcript for FD-SFEM

```
% applying the FORM analysis for a ten-bar truss structure
% a function is defined in matlab which uses the strand 7 API to do a FEM
% analysis of the truss model
% the results of the finite element analysis are called and used in
% calculation of the derivatives of random variables
% Strand7 is used for the FEM analysis
%%%%%%%%%%%%%%%%%%%%%%%Start of the FORM analysis%%%%%%%%%%%%%%%%%%%%%%%
%entering the initial value of the applied load
clear all
clc
% Paramters of the Load effect
muF = 174;
sigmaF = 60.9;
%Parameters of the Resistance
Fy = 350;
muFy = 407;
sigmaFy = 28.5;


kl = 1;
disp(' ')
disp(' ')

disp('..................... Calculating the Resistances and load effect of the ...
    Truss members .....................')
% getting the length and properties of the element truss element
disp('')
disp('')
disp('Calculating the resistances.........')

[leng,proper,modulus] = secprop;
leng1 = leng*1000;%element length
E = modulus*10^-6;
% obtaining areas and moment of inertia in milimeters

Areas = proper(:,1)*10^6;
Inertias = proper(:,3)*(10^12);

% caculating the radii of gyration

gyr = sqrt(transpose(Inertias)./transpose(Areas));

%calculating the mean slenderness ratio for the the members

slnr = ((kl*leng1)./(pi*gyr))*sqrt(Fy/E);
nc = 1.34;
factor = (1+slnr.^(2*nc)).^(-1/nc);

disp('Calculating the tension and compression resistances of all elements....')
```

```
Rt = transpose(Areas).*Fy/1000;%computing the tension resistance of members

Rc = transpose(Areas).*factor*Fy/1000;%calculating compression resistance of ...
    members

n = sum(Rt>0);
number = 1:n;

disp('....End of resistance calculation....')
disp('')
disp('Calculating the internal member forces.........')

[mforce] = tenbartruss([-muF,-0.8*muF]);

for i=1:n

    if mforce(i) < 0
        typ(i) = {'comp'};
    else
        typ(i) = {'tens'};
    end
end

for i = 1:n
    if mforce(i)<0 && abs(mforce(i))> 0.9*Rc(i)

        con(i) = {'fail'};

    elseif mforce(i)>0 && mforce(i)> 0.9*Rt(i)

        con(i) = {'fail'};
    else

        con(i) = {'Ok'};
    end
end




disp('....End of internal load calculation....')
disp('')
disp('..........displaying the results..........')


%matr = [transpose(number),transpose(mforce),transpose(Rt),transpose(slnr),...
    transpose(Rc)];
%horzcat(vertcat({'Beam NO' 'Forces' 'Tr(KN)' 'Slenderness' 'Cr(KN)'},num2cell(...
    matr)))
m1 = [transpose(number),transpose(mforce)];
```

```
m2 =[transpose(Rt),transpose(slnr),transpose(Rc)];
vertcat({'Beam NO' 'Forces' 'type' 'Tr(KN)' 'Slenderness' 'Cr(KN)' 'condition'},...
    horzcat(num2cell(m1),transpose(typ),num2cell(m2),transpose(con)))



disp('..................... End of the resistance and load effect calculation ...
    .....................')
disp('')
disp('')
disp('')
disp('')
disp('..................... Starting FORM Reliability Analysis of the Structure ...
    .....................')

%Creating place holder for the variables
b = zeros(1,n);
g1 = zeros(1,n);
muN = zeros(1,3);
sigmaN = zeros(1,3);
P = zeros(1,3);
g2 = zeros(1,3);
delG = zeros(1,3);
delGN = zeros(1,3);
Pnew = zeros(1,3);
beta = zeros(1,n);
beta1 = zeros(1,n);
g3 = zeros(1,n);
R = zeros(1,n);
%counting the time it takes to do the reliability analysis
tic;
for i = 1:n
    F = muF;

    yield = muFy;%design point for yield stress random variable

    [mforce] = tenbartruss([-F,-0.8*F]);% design point for forces


    if mforce(i)<0

        R(i) = Areas(i)*factor(i)*(1/1000);

    else

        R(i) = Areas(i)*(1/1000);
    end

    R1 = R;
    beta1(i) = 0;
```

```
resBeta = 20;
g3(i) = 100;
abbas =[resBeta,g3(i)];
conv = abbas<0.001;
%forming a while loop to check convergence criteria
mforce1 = mforce;
while abs(resBeta) > 0.001 %sum(conv) < 2

    %check if the member are in tension or compression

    %computing the values of limit state function


    g1(i) = R1(i)*yield - abs(mforce1(i));


    %calculating the equivalent normal parameters for load the variables

    [muNf,sigmaNf] = gumbeleq1(abs(F(1)),muF,sigmaF);

     f = (F - muNf)/sigmaNf ;

    % calculating the equivalent normal paramters for resistance variables


    [muNfy,sigmaNfy] = lognrmeq(yield,muFy,sigmaFy);

    fy = (yield - muNfy)/sigmaNfy;

    %Calculating the partial derivative of the limit state fucntion by
    %the method of finite difference

    % A) Computing the derivatives of the load effect

    percent = 0.01;

    F2 = F + percent;

    mforce2 = tenbartruss([-F2,-0.8*F2]);


    g2(1) = R1(i)*yield - abs(mforce2(i));


    delGf = (g2(1) - g1(i))/percent;

    %calculating the partial derivative in the equivalent normal space

    delGNf = delGf*sigmaNf;
```

```
        % B) calculating the derivateves of resistance

        yield2 = yield + percent;

        g2(2) = R1(i)*yield2 - abs(mforce1(i));

        delGfy = (g2(2) - g1(i))/percent;

        delGNfy = delGfy*sigmaNfy;

        %calculating the coordinates of the new design point

        coef = (1/(delGNfy^2 + delGNf^2))*(f*delGNf + fy*delGNfy - g1(i));

        Pnew = [coef*delGNf,coef*delGNfy];

        %calculating the beta value of the member

        beta(i) = sqrt(sum(Pnew.^2));

        resBeta = beta(i)- beta1(i);

        beta1(i)= beta(i);

        %calculating the coordinates of the new iteration point

        F = muNf + sigmaNf*Pnew(1);

        yield = muNfy + sigmaNfy*Pnew(2);

        mforce1 = tenbartruss([-F,-0.8*F]);

        g3(i) = R1(i)*yield - abs(mforce1(i));

        b(i) = b(i) + 1;
        if b(i) > 25
            disp('The number of iterations exceeds 15');
            break

        end
    abbas =[resBeta,g3(i)];
    conv = abbas<0.0001;

    end

    disp(['End of reliability calculation for member',num2str(i)])

end
t = toc;
mint = floor(t/60);
```

```
sec = floor(t - mint*60);
disp('The Beta values obtained for members are: ');
disp(' ');
disp(beta);
disp(' ')
disp(['Total Number of iteration for FORM analysis: ',num2str(sum(b))]);
disp(' ');
disp(['Elapsed time is: ',num2str(mint),' mins and ',num2str(sec),' secs']);
```

II.6.2.2 Matlab transcript for FSFEM

```
% A program to peform a completely stochastic finite element analysis of
% the structure the force value is a random variable muF = 174 and
% sigamF=60.9 and Mufy = 407 and sigmafy = 28.5

muF = 174;% Mean value of F
sigmaF = 60.9; %Standard deviation of F

muFy = 407; %Mean value of yeild stress
sigmaFy = 28.5; %standard deviation of yield stress

F = muF*ones(10,1);% Initial design point for load effect
yield = muFy*ones(10,1);% Initial design point for yield stress

Rc = [1.22962 0.47776 0.00743 0.01307 0.62325 1.13555 0.31150 0.01767 0.22583...
    0.19073];
Rc = transpose(Rc);
% Member resistances with respect to tension
Rt = [1.91637 0.92928 0.18850 0.18850 1.22648 2.27043 1.0490 0.32987 0.92928...
    0.77833];
Rt = transpose(Rt);

mforce = FEMcodetruss([-F,-0.8*F]);
R = zeros(10,1);
for i = 1:10
    if mforce(i)<= 0
        R(i) = Rc(i);
    else
        R(i) = Rt(i);
    end
end

beta = 20*ones(10,1);
beta1 = zeros(10,1);

res = beta - beta1;
n = 0;
Df = zeros(10,1);
Dfy = zeros(10,1);
```

```
muNf = zeros(10,1);
sigmaNf = zeros(10,1);
muNfy = zeros(10,1);
sigmaNfy = zeros(10,1);
coef = zeros(10,1);
coef1 = zeros(1,10);
coef2 = zeros(1,10);
Y = cell(1,10);
beta_rec = zeros(10,10);
while all(abs(res) >= 0.0001)
    n = n + 1
    G = transpose(R).*transpose(yield) - abs(mforce);

    %Equivalent normal paramters for F

%      for i=1:10
%
%          [muNf(i),sigmaNf(i)] = gumbeleq(abs(F(i)),muF,sigmaF);
%
%      end

    muNf = ones(10,1)*muF;
    sigmaNf = ones(10,1)*sigmaF;

    %Equivalent normal paramters for yield stress

%      for i=1:10
%
%          [muNfy(i),sigmaNfy(i)] = lognrmeq(yield(i),muFy,sigmaFy);
%
%      end

    muNfy = ones(10,1)*muFy;
    sigmaNfy = ones(10,1)*sigmaFy;

    for i =1:10

    A = [-muNf(i)/sigmaNf(i);-muNfy(i)/sigmaNfy(i)];
    B = [1/sigmaNf(i) 0;0 1/sigmaNfy(i)];

    Y{i} = B*[F(i);yield(i)] + A;
    end

    %Calculating the derivatives

    for i=1:10
    m = FEMcodetruss([-1,-0.8]);
    Df(i) = -abs(m(i)*sigmaNf(i));
    Dfy(i) = R(i)*sigmaNfy(i);
```

```
    end
    %Calcualing the N-R coefficients

    for i=1:10
        YY = Y{i};
        coef(i) = (1/(Df(i)^2 + Dfy(i)^2))*(Df(i)*YY(1) + Dfy(i)*YY(2) - G(i));

    end

    coef1 = transpose(coef).*transpose(Df);
    coef2 = transpose(coef).*transpose(Dfy);

    %Calculating reliability index values

    beta = sqrt(coef1.*coef1 + coef2.*coef2);
    beta_rec(:,n) = beta';
    %Computing New Desing points

    F = transpose(transpose(muNf) + transpose(sigmaNf).*coef1);

    yield = transpose(transpose(muNfy) + transpose(sigmaNf).*coef2);
    mm = zeros(1,10);
    for j = 1:10
        ff = [-abs(F(j)),-abs(0.8*F(j))];
        mforce = FEMcodetruss(ff);
        mm(j) = mforce(j);
    end
    mforce = mm;
    res = transpose(beta) - beta1;
    beta1 = transpose(beta);


end
```

The FORM method includes the calculation of the equivalent normal parameters for non-normal random variables. The programs that were developed for this purpose are shown below.

II.6.2.3 Matlab transcripts for the equivalent normal parameters of the lognormal distribution

```
function [mu_eq,sigma_eq] = lognrmeq (x,mu,sigma)
%calculation of the equivalent mean and standard deviation for a lognormal
%distribution fuction

%calculating corrsponding lognormal parameters

sigmaln =sqrt(log(1+((sigma^2)/(mu^2))));
muln = log(mu)-0.5*sigmaln^2;
```

```
h = (log(x)-muln)/sigmaln;
F = normcdf(h,0,1);
f = (1/sqrt(2*pi))*(1/(sigmaln*x))*exp(-0.5*((log(x)-muln)/sigmaln)^2);

%calculating the equivalent normal parameters
l = norminv(F,0,1);
sigma_eq = (1/f)*normpdf(l,0,1);
mu_eq = x - sigma_eq*(l);
```

II.6.2.4 Matlab transcripts for the equivalent normal parameters of the Gumbel distribution

```
function [mu_eq,sigma_eq] = gumbeleq(x,mu,sigma)
% a fuction that defines the equivalent normal mean and standard deviation
% For a gumbel(Extreme type I) distribution

%Gumbel parmeters a and u


a = sqrt((pi^2)/(6*sigma^2));
u = mu - (0.5772/a);

% calculating pdf and cdf values for gumbel

%F = evcdf(x,a,u);
%f = evpdf(x,a,u);
F = exp(-exp(-a*(x-u)));
% if F == 1
%     F = 0.9999999;
% else
%     F = F;
% end

f = a*exp(-a*(x-u))*exp(-exp(-a*(x-u)));

%calculating the equivalent normal variables
l = norminv(F,0,1);
sigma_eq = (1/f)*normpdf(l,0,1);
mu_eq = x - sigma_eq*l;
```

II.6.2.5 Matlab transcript for RSM-DMCS

```
% Monte carlo anlaysis for member for of the 10 bar truss strucuture
% the relationship between the force in the member and the two applied
% forces F1 and F2 is calculated by regression so it is actually a respnses
% surface method
% Deciding on the number of iterations
```

```
n = input('Enter the number of simulations: ');

%Generating values for F1


tic
j = 0;
cf = [1.2767,0.5005,0.1326,0.0995,1.2733,1.1222,1.1278,0.2304,0.1658,0.8342];
cfy = [1.2293,0.4778,0.1885,0.1885,1.2260,1.1352,1.0490,0.3310,0.2257,0.7783];

a1 = rand(1,n);
muF = 174;
sigmaF = 60.9;

aF = sqrt((pi^2)/(6*sigmaF^2));
uF = muF - (0.5772/aF);

F = (log(log(1./a1)) - aF*uF)./-aF;

% ab = norminv(a1,0,1);
%
% F = muF + sigmaF*ab;

% Generating random numbers for yield stress

 mufy = 407;
 sigmafy = 28.5;


 a2 = rand(1,n);
 z = norminv(a2,0,1);

 sigmaln =sqrt(log(1+((sigmafy^2)/(mufy^2))));
 muln = log(mufy)-0.5*sigmaln^2;

 b = muln + sigmaln.*z;
 fy = exp(b);
% a2 = rand(1,n);
% ac = norminv(a2,0,1);
% fy = mufy + sigmafy*ac;

G = zeros(10,n);
nf = zeros(1,10);
for i=1:10

     G(i,:) = cfy(i)*fy - cf(i)*F;

     nf(i) = sum(G(i,:)<0);
end
```

```
for i= 1:10

   mean(i) = sum(G(i,:))/n;
   h = (G(i,:)− mean(i)).^2;
   SD(i) = sqrt((1/(n−1))*sum(h))
end

for i = 1:10
    pf(i) = nf(i)/n
end

mean = mean';
SD = SD';

t = toc;
mint = floor(t/60);
sec = floor(t − mint*60);
% pf = nf/(n);
beta = −norminv(pf,0,1);
disp('Proability of failure is: ')
disp(pf)
disp('The reliabilty index: ')
disp(beta)
disp(['Elapsed time is: ',num2str(mint),' mins and ',num2str(sec),' secs']);
```

II.6.2.6 Matlab transcript for RSM-LHCSMC

```
% Monte carlo anlaysis for member for of the 10 bar truss strucuture
% the relationship between the force in the member and the two applied
% forces F1 and F2 is calculated by regression so it is actually a respnses
% surface method
% Deciding on the number of iterations

n = input('Enter the number of simulations: ');

%Generating values for F1
prob = 1/n:1/(n):1−1/n ;% creating the vector of interval probabilities
rndn = rand(1,n−1)/(n);
prob1 = prob + prob.*rndn;%choosing random representitive value for each interval
rndn2 = rand(1)/(n);
prob2 = [prob(1)− rndn2,prob1];

tic

cf = [1.2767,0.5005,0.1326,0.0995,1.2733,1.1222,1.1278,0.2304,0.1658,0.8342];
cfy = [1.2293,0.4778,0.1885,0.1885,1.2260,1.1352,1.0490,0.3310,0.2257,0.7783];

% a1 = rand(1,n);
muF = 174;
```

```
sigmaF = 60.9;

aF = sqrt((pi^2)/(6*sigmaF^2));
uF = muF - (0.5772/aF);

F = (log(log(1./prob2)) - aF*uF)./-aF;

% ab = norminv(prob2,0,1);
%
% F = muF + sigmaF*ab;

% Generating random numbers for yield stress

 mufy = 407;
 sigmafy = 28.5;


 z = norminv(prob2,0,1);

 sigmaln =sqrt(log(1+((sigmafy^2)/(mufy^2))));
 muln = log(mufy)-0.5*sigmaln^2;

 b = muln + sigmaln.*z;
 fy = exp(b);

% ac = norminv(prob2,0,1);
% fy = mufy + sigmafy*ac;

c1 = randperm(n);
c2 = randperm(n);

G = zeros(n,10);
nf = zeros(1,10);
for i = 1:n

    d1 = c1(i);
    d2 = c2(i);

    E = cf*F(d1);
%     F1 = [-F(d1),-0.8*F(d1)];
%     mforce = FEMcodetruss(F1);
%     E = abs(mforce);
    R = cfy*fy(d2);
    G(i,:)=R-E;
    nf(i,:) = G(i,:)<0;
    i
end
for i= 1:10

   mean(i) = sum(G(:,i))/n;
```

```
   SD(i) = sqrt( sum((G(:,i)- mean(i)).^2)/(n-1));

end

for k = 1:10

    pf(k) = sum(nf(:,k))/n;
end

pf = pf';
t = toc;
mint = floor(t/60);
sec = floor(t - mint*60);
% pf = nf/(n);
beta = -norminv(pf,0,1);
beta = beta';
disp('Proability of failure is: ')
disp(pf)
disp('The reliabilty index: ')
disp(beta)
disp(['Elapsed time is: ',num2str(mint),' mins and ',num2str(sec),' secs']);
```

II.6.2.7 Matlab transcript for DMCS of the whole structure

```
% Direct Monte Carlo analysis of the whole tenbar truss structure using the
% finite element code written

% defining the number of simulations

n = input('Enter the number of simulations: ');
a1 = rand(1,n);
% generating random values for the applied load

muF = 174;
sigmaF = 60.9;

aF = sqrt((pi^2)/(6*sigmaF^2));
uF = muF - (0.5772/aF);

F1 = (log(log(1./a1)) - aF*uF)./-aF;

% ab = norminv(a1,0,1);
%
% F1 = muF + sigmaF*ab;

% Generating random values for the resistance
```

```
mufy = 407;
sigmafy = 28.5;

a2 = rand(1,n);
z = norminv(a2,0,1);

sigmaln =sqrt(log(1+((sigmafy^2)/(mufy^2))));
muln = log(mufy)-0.5*sigmaln^2;

b = muln + sigmaln.*z;
fy = exp(b);

% ac = norminv(a2,0,1);
% fy = mufy + sigmafy*ac;

F2 = [-268,-214.4];
[leng,proper,modulus] = secprop;
leng1 = leng*1000;%element length
E = modulus*10^-6;
% obtaining areas and moment of inertia in milimeters

Areas = proper(:,1)*10^6;
Inertias = proper(:,3)*(10^12);

% caculating the radii of gyration

gyr = sqrt(transpose(Inertias)./transpose(Areas));

%calculating the mean slenderness ratio for the the members
kl = 1;
Fy = 350;
slnr = ((kl*leng1)./(pi*gyr))*sqrt(Fy/E);
nc = 1.34;
factor = (1+slnr.^(2*nc)).^(-1/nc);

[mforce2,nb] = tenbartruss(F2);
Rf = zeros(1,nb);

for j = 1:nb

    if mforce2(j) > 0
        Rf(j) = Areas(j)/1000;
    else
        Rf(j) = Areas(j)*factor(j)/1000;
    end
end


%starting the simulation
E1 = zeros(n,10);
```

```
R = zeros(n,10);
G = zeros(n,10);
nf = zeros(n,10);
for i = 1:n

   %forming the vector of applied force


   F = [-F1(i),-0.8*F1(i)];

   mforce = FEMcodetruss(F);
   E1(i,:) = mforce;
   R(i,:) = Rf*fy(i);
   G(i,:) = R(i,:) - abs(E1(i,:));

   %nf(i,:) = G<0;
   nf(i,:) = G(i,:)<0;
   i
%    xlswrite('Results1',[i,nf],sprintf('A%d:K%d',[i,i]))

end

for i = 1:10

    pf(i) = sum(nf(:,i))/n;
end
for i= 1:10

   mean(i) = sum(G(:,i))/n;

   SD(i) = sqrt( sum((G(:,i)- mean(i)).^2)/(n-1));
end
mean = mean'
SD = SD'
beta = -norminv(pf,0,1)
```

II.6.2.8 Matlab transcript for LHCSMC of the whole structure

```
% calculating the reliability of memeber using Latin Hyper cube Sampling
% Linking to Ten-bar truss finite element code
% Sepehr Hashemolhosseini
% November 2012

% Determining the number of intervals
F2 = [-174,-139.2]
n = input('Enter the number of intervals: ')

prob = 1/n:1/(n):1-1/n ;% creating the vector of interval probabilities
rndn = rand(1,n-1)/(n);
```

```
prob1 = prob + prob.*rndn;%choosing random representitive value for each interval
rndn2 = rand(1)/(n);
prob2 = [prob(1)- rndn2,prob1];
%Creating a vector of corresponding probability for the interval
%Corresponding vector for force
muF = 174;
sigmaF = 60.9;


aF = sqrt((pi^2)/(6*sigmaF^2));
uF = muF - (0.5772/aF);

F1 = (log(log(1./prob2)) - aF*uF)./-aF;

% ab = norminv(prob2,0,1);
%
% F1 = muF + sigmaF*ab;

%Correspondig vector for resistance

mufy = 407;
sigmafy = 28.5;


z = norminv(prob2,0,1);

sigmaln =sqrt(log(1+((sigmafy^2)/(mufy^2))));
muln = log(mufy)-0.5*sigmaln^2;

b = muln + sigmaln.*z;
fy = exp(b);

% ac = norminv(prob2,0,1);
%
% fy = mufy + sigmafy*ac;

% Creating a vector to randomly choose reperesentitive values

c1 = randperm(n);
c2 = randperm(n);

%%%%%%%%%%%%%%%calculating the resistance factors for member%%%%%%%%%%%%%

[leng,proper,modulus] = secprop;
leng1 = leng*1000;%element length
E = modulus*10^-6;
% obtaining areas and moment of inertia in milimeters

Areas = proper(:,1)*10^6;
Inertias = proper(:,3)*(10^12);
```

```
% caculating the radii of gyration

gyr = sqrt(transpose(Inertias)./transpose(Areas));

%calculating the mean slenderness ratio for the the members
kl = 1;
Fy = 350;
slnr = ((kl*leng1)./(pi*gyr))*sqrt(Fy/E);
nc = 1.34;
factor = (1+slnr.^(2*nc)).^(-1/nc);

[mforce2,nb] = tenbartruss(F2);
Rf = zeros(1,nb);

for j = 1:nb

    if mforce2(j) > 0
        Rf(j) = Areas(j)/1000;
    else
        Rf(j) = Areas(j)*factor(j)/1000;
    end
end
% nb =10
% Rf = [1.1151,0.4024,0.2482,0.1885,1.3580,1.2594,0.9390,0.4876,0.2931,0.6597];
tic;
% Starting to check the simulated values
% nf = zeros(n,nb);
G = zeros(n,10);
nf = zeros(n,10);
for i = 1:n

   %forming the vector of applied force
   d1 = c1(i);
   d2 = c2(i);

   F = [-F1(d1),-0.8*F1(d1)];

   mforce = FEMcodetruss(F);
   E = mforce;
   R = Rf*fy(d2);
   G(i,:) = R - abs(E);

   %nf(i,:) = G<0;
   nf(i,:) = G(i,:)<0;
   i
%    xlswrite('Results',[i,nf],sprintf('A%d:K%d',[i,i]))

end
```

```
% Calculating the failure probabilities for each element
pf = zeros(1,nb);
for k = 1:nb

    pf(k) = sum(nf(:,k))/n;
end

%calculaing the mean and standard deviation
mean = zeros(1,nb);
SD = zeros(1,nb);
for i= 1:nb

   mean(i) = sum(G(:,i))/n;

   SD(i) = sqrt( sum((G(:,i)- mean(i)).^2)/(n-1));
end
t = toc;
hr = floor(t/3600);
mint = floor((t- hr*3600)/60);
sec = floor(t - hr*3600 - mint*60);
disp(['Elapsed time is: ',num2str(hr),' hours ',num2str(mint),' mins and ',...
    num2str(sec),' secs']);
disp('')
beta = -norminv(pf,0,1)
beta = beta';
mean = mean';
SD = SD';
pf = pf';
% disp('The probabilities of failure obtained for members are: ');
% disp(' ');
 disp(pf);
%
% disp('The reliability indices obtained for members are: ');
% disp(' ');


function [mu_eq,sigma_eq] = lognrmeq (x,mu,sigma)
%calculation of the equivalent mean and standard deviation for a lognormal
%distribution fuction

%calculating corrsponding lognormal parameters

sigmaln =sqrt(log(1+((sigma^2)/(mu^2))));
muln = log(mu)-0.5*sigmaln^2;
h = (log(x)-muln)/sigmaln;
F = normcdf(h,0,1);
f = (1/sqrt(2*pi))*(1/(sigmaln*x))*exp(-0.5*((log(x)-muln)/sigmaln)^2);

%calculating the equivalent normal parameters
l = norminv(F,0,1);
```

```
sigma_eq = (1/f)*normpdf(1,0,1);
mu_eq = x - sigma_eq*(1);
```

II.6.2.9 Matlab transcript for ULHCSMC of the whole structure

```
% Performing Updated system sampling Latin Hypercuge Monte Carlo Simulation
% for the 10 bar-truss structure Using the written Finite Element Code
% Sepehr Hashemolhosseini
% November  2012
% Universiteit van Stellenbosch

n = input('Enter the number of intervals: ')
F2 = [-174,-139.2]
prob = 1/n:1/(n):1-1/n ;% creating the vector of interval probabilities
rndn = rand(1,n-1)/(n);
prob1 = prob + prob.*rndn;%choosing random representitive value for each interval
rndn2 = rand(1)/(n);
prob2 = [prob(1)- rndn2,prob1];

P1 = randperm(n)';
P2 = randperm(n)';

% Starting Updated Sampling Procedure
for k = 1:3

    P = [P1,P2];
    b = size(P);
    h = b(2);
    T = zeros(h);
    for i = 1:h
      for j = 1:h
         kn = 6*sum((P(:,i)-P(:,j)).^2);
         T(i,j)= 1 - (kn/(n*(n-1)*(n+1)));
      end
    end

    Q = chol(T,'lower');%Cholesky factorization of the matrix
    S = inv(Q);% inverse of the cholesky partition
    Ps = P*transpose(S);

    p1 = Ps(:,2);
    p2 = zeros(n,1);
    p_s = sort(p1);
    pj = 1:n;
    pj = pj';

    %rearranging the permutation marix

    for i = 1:n
```

```
        for j = 1:n
           if p1(i) == p_s(j)
              p2(i) = pj(j);
           end
        end
        i
     end

     P = [P1,p2];
end
% Generating random numbers for load variables
muF = 174;
sigmaF = 60.9;

% aF = sqrt((pi^2)/(6*sigmaF^2));
% uF = muF - (0.5772/aF);
%
% F1 = (log(log(1./prob2)) - aF*uF)./-aF;

ab = norminv(prob2,0,1);

F1 = muF + sigmaF*ab;

% Generating random numbers for yield stress
mufy = 407;
sigmafy = 28.5;

% z = norminv(prob2,0,1);
%
% sigmaln =sqrt(log(1+((sigmafy^2)/(mufy^2))));
% muln = log(mufy)-0.5*sigmaln^2;
%
% b = muln + sigmaln.*z;
% fy = exp(b);

ac = norminv(prob2,0,1);

fy = mufy + sigmafy*ac;

c1 = P(:,1)';
c2 = P(:,2)';
%%%%%%%%%%%%%%%calculating the resistance factors for member%%%%%%%%%%%%%
[leng,proper,modulus] = secprop;
leng1 = leng*1000;%element length
E = modulus*10^-6;
% obtaining areas and moment of inertia in milimeters

Areas = proper(:,1)*10^6;
Inertias = proper(:,3)*(10^12);
```

```
% caculating the radii of gyration

gyr = sqrt(transpose(Inertias)./transpose(Areas));

%calculating the mean slenderness ratio for the the members
kl = 1;
Fy = 350;
slnr = ((kl*leng1)./(pi*gyr))*sqrt(Fy/E);
nc = 1.34;
factor = (1+slnr.^(2*nc)).^(-1/nc);

[mforce2,nb] = tenbartruss(F2);
Rf = zeros(1,nb);

for j = 1:nb

    if mforce2(j) > 0
        Rf(j) = Areas(j)/1000;
    else
        Rf(j) = Areas(j)*factor(j)/1000;
    end
end

tic;
% Starting to check the simulated values
% nf = zeros(n,nb);
G = zeros(n,10);
nf = zeros(n,10);
for i = 1:n

   %forming the vector of applied force
   d1 = c1(i);
   d2 = c2(i);

   F = [-F1(d1),-0.8*F1(d1)];

   mforce = FEMcodetruss(F);
   E = mforce;
   R = Rf*fy(d2);
   G(i,:) = R - abs(E);

   %nf(i,:) = G<0;
   nf(i,:) = G(i,:)<0;
   i


end

% Calculating the failure probabilities for each element
pf = zeros(1,nb);
```

```
for k = 1:nb

    pf(k) = sum(nf(:,k))/n;
end

%calculaing the mean and standard deviation
mean = zeros(1,nb);
SD = zeros(1,nb);
for i= 1:nb

   mean(i) = sum(G(:,i))/n;

   SD(i) = sqrt( sum((G(:,i)- mean(i)).^2)/(n-1));
end
t = toc;
hr = floor(t/3600);
mint = floor((t- hr*3600)/60);
sec = floor(t - hr*3600 - mint*60);
disp(['Elapsed time is: ',num2str(hr),' hours ',num2str(mint),' mins and ',...
    num2str(sec),' secs']);
disp('')
beta = -norminv(pf,0,1)
beta = beta';
mean = mean';
SD = SD';
pf = pf';
disp('The probabilities of failure obtained for members are: ');
disp(' ');
disp(pf);

disp('The reliability indices obtained for members are: ');
disp(' ');
disp(beta)
```

II.7 Transcripts for the system reliability evaluation

Two methods were used for the system reliability evaluation: the β-unzipping method and the branch and bound method. The programs for these methods are presented in sections below.

II.7.1 Transcripts for the β-unzipping system reliability analysis of the 10-bar truss structure

For the system reliability analysis using the β-unzipping approach, 6 programs are developed. One general program, 2 damaged state programs for the damaged state analyses at levels 2 and 3, 2 system reliability programs for the system reliability analyses of the triples of elements at level 3 and pairs of elements at level 2, and one reliability analysis program for the component reliability analysis of the intact structure as well as the damaged structure.

II.7.1.1 Matlab transcript of the general system reliability program

```
%General Module for modeling the structure and calculation of the system
%reliability up to level 3

% Defining the material properties and applied loads

F = [-1,-0.8];

E = 2*10^11;
A = [0.001961,0.000929,0.000188,0.000188,0.001226,0.00227,0.001049,0.000331,...
    0.000929,0.000778];
% Creating the geometrical Model

numberElements = 10;
numberNodes = 6;

elementNodes = [1 2;2 3;3 6;5 6;4 5;1 5;2 4;2 5;2 6;3 5];
nodeCoordinates = [0 0;3 0;6 0;0 4;3 4;6 4];

% Calculatin the orientaion of the elements
theta = zeros(0,numberElements);%theta, vector of element orientations

for i=1:numberElements
    a = elementNodes(i,:);
    b = nodeCoordinates(a(2),:);
    c = nodeCoordinates(a(1),:);
    delta = b' - c';
    theta(i) = atan(delta(2)/delta(1));
end

%member resistances with respect to compression

Rc = [1.22962 0.47776 0.00743 0.01307 0.62325 1.13555 0.31150 0.01767 0.22583...
    0.19073];
Rc = transpose(Rc);
% Member resistances with respect to tension
Rt = [1.91637 0.92928 0.18850 0.18850 1.22648 2.27043 1.0490 0.32987 0.92928...
    0.77833];
Rt = transpose(Rt);

xx = nodeCoordinates(:,1);
yy = nodeCoordinates(:,2);
% Forming the displacement vector, load vector and stiffness matrix

GDof = 2*numberNodes;
```

```
displacements = zeros(GDof,1);
force = zeros(GDof,1);

% applying the load on the structure

force(4) = F(1)*1000;
force(6) = F(2)*1000;

[stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A);

% Difnining boundry conditons

prescribedDof = [1 2 7 8];

% calculating the displacements stresses and intenal forces
displacements = solution(GDof,prescribedDof,stiffness,force);

[sigma,mforce] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A);

 mforce = mforce/1000;

 %%%%%%%%%%%%%%%% Calculating the reliabiilty at level 1%%%%%%%%%%%%%%%%%%

 result1 = zeros(numberElements,4);
 mode = zeros(numberElements,1);
 for i = 1:numberElements
     if mforce(i)<0
         [beta,alpha_fy,alpha_F] = relanalysis([Rc(i),mforce(i)]);
         mode(i) = 0; %Failure in compression

     else
         [beta,alpha_fy,alpha_F] = relanalysis([Rt(i),-mforce(i)]);
         mode(i) = 1; %Failure in Tension
     end
     result1(i,:) = [i,beta,alpha_fy,alpha_F];
 end
 matrix1 = [result1,mode]; % Matrix including the info about intact structure
 beta = matrix1(:,2);
 disp ('the results of the reliability evaluation for the intact structure: ')
 disp('')
 disp(matrix1)

 disp('Minimum beta value is :')
 disp(min(beta))
 Mbeta = min(beta);

 deltB = input('Enter the value of delta beta for unzipping at level 1 :')
```

```
% identifying critical members accroding to deltaB
L=0;
for i =1:numberElements
    if beta(i)<=(Mbeta + deltB)
        L = L + 1;
        crit1(L,:) = matrix1(i,:);
    end
end
critical = crit1(:,1); %List of the critical components (Failure Elementst)with ...
    the corresponding information
disp(' The critical memebers are :')
disp(critical)
% here the results of crit1 need to be sent to the system reliability
% module to calculate the system reliability at level 1 which is a series
% system

% Calculating the correlation matrix
alpha = crit1(:,[3,4]);
n = size(alpha);
corln = zeros(n(1));
for i = 1:n(1)
   for j = 1: n(1)
       if j==i
           corln(i,j) = 1;
       else
           cheragh = alpha(i,:)*transpose(alpha(j,:));
           corln(i,j)= cheragh;
           if abs(cheragh-1)> 1e-15
               corln(i,j)=0.9999;
           end
       end
   end
end

disp('the correlation matrix at level 1 is :')
disp(corln)

% forming Ditlevsen Bounds for the systme
lower_h = normcdf(-crit1(:,2));
lower = max(lower_h);
upper_h = normcdf(crit1(:,2));
upper = 1 - prod(upper_h);
disp (' the Ditlevsen bound for the system failure probability at level one are ...
    :')
disp (['the lower bound is :',num2str(lower)])
disp (['the upper bound  is :',num2str(upper)])
disp('Ditlevsen bounds for the system reliability at level one  are :')
% shown the reliability index for the bounds
disp (['The Upper RI Bound Is :',num2str(-norminv(lower))])
disp (['The Lower RI Bound  Is :',num2str(-norminv(upper))])
```

```
 %%%%%%%%%%%%%%%%%%%System reliability at level 2%%%%%%%%%%%%%%%%%%%%%%%%%


 % Calculating system reliability at level 2

 lz = 0;
 critical = transpose(critical);
 for i = critical
      lz = lz + 1
     [members,betas,alphas,modes,aFy] = Damaged(A,E,mode(i),numberElements,...
         numberNodes,elementNodes,...
         nodeCoordinates,Rc,Rt,critical(lz),theta(i));

      mat = [members,betas,alphas,modes];
%     matrix = vertcat(matrix,mat);
      matrix{lz}= mat; % a cell array including the data about each damaged state
      factors{lz}= aFy;
 end
 % identifiying the critical elements at level 2
hh = 0;
kk = 0;
syze = size(critical);
Mbeta2 = zeros(syze(2),syze(1));

% defninig the minimum reliability index values
 for i = critical
     kk = kk + 1;
     M = matrix{kk};
     Mbeta2(kk) = min(M(:,2)); % vector of minimum beta values

 end

 deltB2 = input('Enter the value of delta beta for unzzipping at level 2 :');


  %identifying member numbers that are critical
  kk1 = 0;
  LL =0;
  for i = critical
      kk1 = kk1 + 1;
      MB = matrix{kk1};
      for j = 1:(numberElements-1)
          if MB(j,2)<= (Mbeta2(kk1)+ deltB2)
              LL = LL+1;
              crit(LL,:)= MB(j,:);
          end
          sz = size(crit);
          %checking if the rows of crit is different in each loop
```

```
        end
        % removing rows left from the previous loops
        if  LL<sz(1)
            crit = crit(1:LL,:);
        end
        crit2{kk1} = crit;
        LL = 0;
        MB = 0;
    end
disp('Critical elements at level 2 are as shown below :')
hh1 = 0;
for i = critical
   hh1= hh1 + 1;
   disp(['the critical elements for the damaged state of the failure of member',...
       num2str(critical(hh1))]);
   crit2{hh1}
end

% Showing the list of critical failure elements pairs and their failure
% modes
hh4 = 0;
hh2 = 0;
hh3 = 0;
for i = critical
    hh2 = hh2 + 1;
    P_H = crit2{hh2};
    P_H2 = crit1(hh2,:);
    critical2 = P_H(:,1);
    critical2 = transpose(critical2);
    for j = critical2
        hh4 = hh4 + 1;
        hh3 = hh3 + 1;
        pairs(hh3,:) = [i,j];
        modes2(hh3,:) = [P_H2(5),P_H(hh4,5)];
        Data2(hh3,:) = [P_H2(2),P_H2(3),P_H2(4),P_H(hh4,2),P_H(hh4,3),P_H(hh4,4)...
            ]; % storing data about critical elements at level 2
    end
    hh4 = 0;
end

% Calculating the system reliability

[pair_s,beta_s,alphafy_s,alphaF_s] = Syslev2(crit1,crit2);
disp([pair_s,beta_s,alphafy_s,alphaF_s])

alpha2 = [alphafy_s,alphaF_s];

n2 = size(alpha2);
corln2 = zeros(n2);
 for i = 1:n2(1)
```

```
    for j = 1: n2(1)
        if i==j
            corln2(i,j) = 1;
        else
            cheragh2 = alpha2(i,:)*transpose(alpha2(j,:));
            corln2(i,j)= cheragh2;
            if abs(cheragh2-1)>1e-15
                corln2(i,j)=0.9999;
            end
        end
    end
  end

disp('the correlation matrix at level 2 is :')
disp(corln2)

%forming the Ditlevsen bounds for the system

lower_h2 = max(normcdf(-beta_s));
upper_h2 = 1 - prod(normcdf(beta_s));
disp (' the Ditlevsen bound for the system reliability at level two are :')
disp (['the lower bound is :',num2str(lower_h2)])
disp (['the upper bound  is :',num2str(upper_h2)])
disp (['The Upper RI Bound Is :',num2str(-norminv(lower_h2))])
disp (['The Lower RI Bound  Is :',num2str(-norminv(upper_h2))])

 % calculating the reliability of pairs

 %%%%%%%%%%%%%%%%% System reliability at level 3%%%%%%%%%%%%%%%%%%%%%%%%%


 s2 = size(pairs);
 for i = 1:s2(1)
     i
     pairs3 = pairs(i,:)
     thetas = [theta(pairs3(1)),theta(pairs3(2))];
     [members,betas,alphas,modes] = Damaged2(A,E,modes2(i,:),numberElements,...
         numberNodes,elementNodes,...
         nodeCoordinates,Rc,Rt,pairs3,thetas);
      mat3 = [members,betas,alphas,modes];
%     matrix = vertcat(matrix,mat);
      matrix3{i}= mat3; % a cell array including the data about each damaged ...
          state
 end

kk = 0;
syze1 = size(pairs);
Mbeta3 = zeros(syze(1),1);

% defninig the minimum reliability index values
```

```
 for i = 1:syze1(1)
     kk = kk + 1;
     M3 = matrix3{kk};
     Mbeta3(kk) = min(M3(:,2)); % vector of minimum beta values
 end

 deltB3 = input('Enter the value of delta beta for unzzipping at level 3 :');

%identifying member numbers that are critical
kk1 = 0
LL = 0
syze2 = size(pairs)
 for i = 1:syze2(1)
      kk1 = kk1 + 1
      MB3 = matrix3{kk1}
      for j = 1:(numberElements-2)
          if MB3(j,2)<= (Mbeta3(kk1)+ deltB3)
              LL = LL+1
              crit_3(LL,:)= MB3(j,:)
          end
      end
      sz2 = size(crit_3);
      %removing rows left from previous loops
      if LL<sz2(1)
          crit_3 = crit_3(1:LL,:);
      end
      crit3{kk1} = crit_3; %cell array holding the critical states information
      LL = 0;
      MB3 = 0;
 end

 % displaying the critical triple of fialure elements pairs

 disp('Critical elements at level 3 are as shown below :')
 disp(pairs)
 hh2 = 0;
 hh3 = 0;
  for i = 1:syze2(1)
     hh2 = hh2 + 1;
     P_H3 = crit3{hh2};
     critical3 = P_H3(:,1);
     for j = critical3'
         hh3 = hh3 + 1;
         triples(hh3,:) = [pairs(hh2,:),j];
     end
  end

 % calculating the system reliability at level 3

 [beta_s3,alphafy_s3,alphaF_s3] = Syslev3(Data2,crit3,delt);
```

```
 % showing the reliability index for each triple of failure elements
 disp('    EL. 1      EL.2      EL.3       RI')
 disp([triples,beta_s3])

 alpha3 = [alphafy_s3,alphaF_s3];
 n3 = size(alpha3);
 corln3 = zeros(n3);
  for i = 1:n3(1)
    for j = 1: n3(1)
        if i==j
            corln3(i,j) = 1;
        else
            corln3(i,j)= alpha3(i,:)*transpose(alpha3(j,:));
            if abs(corln3(i,j)-1)>=1e-15
                corln3(i,j)=0.9999;
            end
        end
    end
  end

disp('the correlation matrix at level 3 is :')
disp(corln3)
%forming the Ditlevsen bounds for the system

lower_h3 = max(normcdf(-beta_s3));
upper_h3 = 1 - prod(normcdf(beta_s3));
disp (' the Ditlevsen bound for the system reliability at level two are :')
disp (['the lower bound is :',num2str(lower_h3)])
disp (['the upper bound  is :',num2str(upper_h3)])
```

II.7.1.2 Matlab transcript of the damaged state program at level 2

```
% This is the damaged module of the structure, it evaluates the damaged
% state of the structure by removing the members modifiying the strucutre
% inputs of the function are A (vector of member areas, number of elements
% element nodes,...
% it gives the members togehter with their corresponding reliability indices
% their sensitivity factors and the goverinig failure mode ("1" for tension
% and "0" for compression)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%inputs %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [members,betas,alphas,modes,aFy] = Damaged(A,E,mode,numberElements,...
    numberNodes,elementNodes,...
    nodeCoordinates,Rc,Rt,critical,theta)

A = transpose(A);

% Rc %vector of compression resistances
% Rt %vector of tension resistances
```

```
GDof = 2*numberNodes;
force = zeros(GDof,1);
force1 = zeros(GDof,1);
members = zeros(numberElements,1);
for i=1:numberElements
    members(i)= i;
end

disp(theta)
%failure mode of the critical element Tension =1, Compresson =0


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%critical element node numbers
Nodes = elementNodes(critical,:);

%Forming the Degrees of freedom for the critical element

Dofs = [Nodes(1)*2-1;Nodes(1)*2;Nodes(2)*2-1;Nodes(2)*2];

%removing the area of the critical element form the vector of areas

A = removerows(A,'ind',critical);
members = removerows(members,'ind',critical);


%removing the element node numbers from the matrix of element nodes

elementNodes = removerows(elementNodes,'ind',critical);

% removing the critical elements resistance from the resistances vectors

Rc1 = removerows(Rc,'ind',critical);% resistance of critical element removed
Rt1 = removerows(Rt,'ind',critical);%resistance of critical element removed

%modifining the number of elements

numberElements = numberElements - 1;

% forming the signs vector for the elements

if theta>=0
     if mode == 1
         aL = [cos(theta) sin(theta) -cos(theta) -sin(theta)];
     else
         aL = [-cos(theta) -sin(theta) cos(theta) sin(theta)];
     end
else
```

```
        if mode == 1
            aL =[-cos(theta) -sin(theta) cos(theta) sin(theta)];
        else
            aL = [cos(theta) sin(theta) -cos(theta) -sin(theta)];
        end
end
disp(aL)
 % modifing the global force vector including the fictitious loads

 for i=1:4
     force1(Dofs(i)) = aL(i);
 end


displacements = zeros(GDof,1);

 % applying the external loads on the damaged structure
force(4) = -1000;
force(6) = -0.8*1000;

xx = nodeCoordinates(:,1);
yy = nodeCoordinates(:,2);

[stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A);

% Difnining boundry conditons

prescribedDof = [1 2 7 8];

% Calculating the influence factor with respect to the external load
displacements = solution(GDof,prescribedDof,stiffness,force);

[sigma,aF] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A);
aF = aF/1000
disp(aF)%%%%%%%

% Calculation the influence factors with respect to the post-failure
% capacity of the critical element

displacements1 = solution(GDof,prescribedDof,stiffness,force1);

[sigma,aFy] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements1,E,A);

if mode == 1
    aFy = Rt(critical)*aFy;
else
    aFy = Rc(critical)*aFy;
```

```
end
aFy = aFy';
disp(aFy)%%%%%%%%%%%%%%%%%
%Forming the new resistance coefficients
Rt2 = Rt1 - aFy %Resistance factors at level 2 for tension
Rc2 = Rc1 + aFy %Resistance factors at level 2 for compression

% Sending the aF, Rt2 and Rc2 to the reliability analysis program
% Calculation of reliability indices with respect to Rt2
betas_t = zeros(1,numberElements);
alf_fyt = zeros(1,numberElements);
alf_Ft = zeros(1,numberElements);
m = 0
for i = 1:numberElements
    m = m+1
    vect = [Rt2(i),-aF(i)];
    [betas_t(i),alf_fyt(i),alf_Ft(i)] = relanalysis(vect);
end

% calculation of reliability indices with respect to compression failure

betas_c = zeros(1,numberElements);
alf_fyc = zeros(1,numberElements);
alf_Fc = zeros(1,numberElements);
nn = 0
for i = 1:numberElements
    nn = nn + 1
    vect = [Rc2(i),aF(i)];
    [betas_c(i),alf_fyc(i),alf_Fc(i)] = relanalysis(vect);
end

% identifiyin the dominant failure modes
betas = zeros(numberElements,1);
alphas = zeros(numberElements,2);
modes = zeros(numberElements,1);
for i = 1:numberElements

    betas(i) = min(betas_c(i),betas_t(i))
    if betas_c(i)<betas_t(i)
        alphas(i,:)= [alf_fyc(i),alf_Fc(i)]
        modes(i)=0
    else
        alphas(i,:)= [alf_fyt(i),alf_Ft(i)]
        modes(i)=1
    end
end
disp(' RI  alphafy   alphaF   mode(1=t 0=c)')

end
```

II.7.1.3 Matlab transcript of the damaged state program at level 3

```
% This is the damaged module of the structure, it evaluates the damaged
% state of the structure at level by removing the members modifiying the ...
    strucutre
% inputs of the function are A (vector of member areas, number of elements
% element nodes,
% it give the members togehter with their corresponding reliability indices
% their sensitivity factors and the goverinig failure mode ("1" for tension
% and "0" for compression)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%inputs %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [members,betas,alphas,modes,aFy] = Damaged2(A,E,mode,numberElements,...
    numberNodes,elementNodes,...
     nodeCoordinates,Rc,Rt,critical,theta)

A = transpose(A);
% Rc %vector of compression resistances
% Rt %vector of tension resistances
GDof = 2*numberNodes;
force = zeros(GDof,1);
force1 = zeros(GDof,1);
force2 = zeros(GDof,1);
members = zeros(numberElements,1);
for i=1:numberElements
    members(i)= i;
end
disp(theta)
%failure mode of the critical element Tension =1, Compresson =0

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%critical element node numbers
Nodes = elementNodes(critical,:);
Nodes1 = Nodes(1,:);
Nodes2 = Nodes(2,:);

%Forming the Degrees of freedom for the critical element

Dofs1 = [Nodes1(1)*2-1;Nodes1(1)*2;Nodes1(2)*2-1;Nodes1(2)*2]
Dofs2 = [Nodes2(1)*2-1;Nodes2(1)*2;Nodes2(2)*2-1;Nodes2(2)*2]

%removing the area of the critical element form the vector of areas

A = removerows(A,'ind',critical);
members = removerows(members,'ind',critical);


%removing the element node numbers from the matrix of element nodes
```

```
elementNodes = removerows(elementNodes,'ind',critical);

% removing the critical elements resistance from the resistances vectors

Rc1 = removerows(Rc,'ind',critical);% resistance of critical element removed
Rt1 = removerows(Rt,'ind',critical);%resistance of critical element removed

%modifining the number of elements

numberElements = numberElements - 2;

% forming the signs vector for the elements and applying the unit load for
% the first failure element

if theta(1)>=0
     if mode(1) == 1
         aL1 = [cos(theta(1)) sin(theta(1)) -cos(theta(1)) -sin(theta(1))]
     else
         aL1 = [-cos(theta(1)) -sin(theta(1)) cos(theta(1)) sin(theta(1))]
     end
else
     if mode(1) == 1
         aL1 =[-cos(theta(1)) -sin(theta(1)) cos(theta(1)) sin(theta(1))]
     else
         aL1 = [cos(theta(1)) sin(theta(1)) -cos(theta(1)) -sin(theta(1))]
     end
end
disp(aL1)
 % modifing the global force vector including the fictitious loads

 for i=1:4
     force1(Dofs1(i)) = aL1(i);
 end

 % forming the signs vector for the elements and applying the unit load for
 % the second failure element

 if theta(2)>=0
     if mode(2) == 1
         aL2 = [cos(theta(2)) sin(theta(2)) -cos(theta(2)) -sin(theta(2))]
     else
         aL2 = [-cos(theta(2)) -sin(theta(2)) cos(theta(2)) sin(theta(2))]
     end
else
     if mode(2) == 1
         aL2 = [-cos(theta(2)) -sin(theta(2)) cos(theta(2)) sin(theta(2))]
     else
         aL2 = [cos(theta(2)) sin(theta(2)) -cos(theta(2)) -sin(theta(2))]
     end
```

```
end
disp(aL2)
% modifiy creating a force matrix for the fictitious load with respect to
% the second failed member
 for i=1:4
     force2(Dofs2(i)) = aL2(i);
 end

displacements = zeros(GDof,1);

 % applying the external loads on the damaged structure
force(4) = -1000;
force(6) = -0.8*1000;


xx = nodeCoordinates(:,1);
yy = nodeCoordinates(:,2);

[stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A);

% Difnining boundry conditons

prescribedDof = [1 2 7 8];

% Calculating the influence factor with respect to the external load
displacements = solution(GDof,prescribedDof,stiffness,force);

[sigma,aF] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A);
aF = aF/1000
disp(aF)%%%%%%%

% Calculation the influence factors with respect to the post-failure
% capacity of the first critical element

displacements1 = solution(GDof,prescribedDof,stiffness,force1);

[sigma,aFy1] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements1,E,A);

%calculation of the influence factors with respect to the post-failure
%capacity of the second failure element
displacements2 = solution(GDof,prescribedDof,stiffness,force2);
[sigma,aFy2] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements2,E,A);

%applying the post-failure capacity to the influence factors
%first critical member
if mode(1) == 1
```

```
    aFy1 = Rt(critical(1))*aFy1;
else
    aFy1 = Rc(critical(1))*aFy1;
end
aFy1 = aFy1'

%Second critical member
if mode(2) == 1
    aFy2 = Rt(critical(2))*aFy2;
else
    aFy2 = Rc(critical(2))*aFy2;
end
aFy2 = aFy2'

%Forming the new resistance coefficients
Rt2 = Rt1 - aFy1 - aFy2 %Resistance factors at level 2 for tension
Rc2 = Rc1 + aFy1 + aFy2 %Resistance factors at level 2 for compression

% Sending the aF, Rt2 and Rc2 to the reliability analysis program
% Calculation of reliability indices with respect to tension failure Rt2
betas_t = zeros(1,numberElements);
alf_fyt = zeros(1,numberElements);
alf_Ft = zeros(1,numberElements);
m = 0
for i = 1:numberElements
    m = m+1
    vect = [Rt2(i),-aF(i)];
    [betas_t(i),alf_fyt(i),alf_Ft(i)] = relanalysis(vect);
end
% calculation of reliability indices with respect to compression failure
% Rc2

betas_c = zeros(1,numberElements);
alf_fyc = zeros(1,numberElements);
alf_Fc = zeros(1,numberElements);
nn = 0
for i = 1:numberElements
    nn = nn + 1
    vect = [Rc2(i),aF(i)];
    [betas_c(i),alf_fyc(i),alf_Fc(i)] = relanalysis(vect);
end

% identifiyin the dominant failure modes
betas = zeros(numberElements,1);
alphas = zeros(numberElements,2);
modes = zeros(numberElements,1);
for i = 1:numberElements

    betas(i) = min(betas_c(i),betas_t(i))
    if betas_c(i)<betas_t(i)
```

```
        alphas(i,:)= [alf_fyc(i),alf_Fc(i)]
        modes(i)=0
    else
        alphas(i,:)= [alf_fyt(i),alf_Ft(i)]
        modes(i)=1
    end
end
disp(' RI  alphafy   alphaF   mode(1=t 0=c)')

end
```

II.7.1.4 Matlab transcript of the parallel systems reliability analysis program at level 2

```
% A function to calculate the system reliability of each pair and their
% equivalent saftey margin sensitivity vectors
% it outputs a vector of RI values, resistance sensitivity factors and
% load effect sensitivity facctors

function [pair_s,beta_s,alphafy_s,alphaF_s] = Syslev2(crit1,crit2)
h = 0;
n1 = size(crit1);
for i = 1:n1(1)
    M = crit1(i,:);
    M1 = crit2{i};
    n2 = size(M1);
    for j = 1:n2(1)
        h = h + 1;
        M2 = M1(j,:);
        roh = M(3)*M2(3) + M(4)*M2(4);
        if abs(roh-1)> 1e-15
            roh = 0.9999;
        end
        mu = [0,0];
        sigma = [1 roh;roh 1];

        KK = mvncdf([-M(2),-M2(2)],mu,sigma); %RI for the pair
        beta_s(h) = -norminv(KK);

        %calculating the equivalent sensitivy factors

        eps1 = [0.1;0];
        eps2 = [0;0.1];

        beta1 = -[M(2);M2(2)] - [M(3) M(4);M2(3) M2(4)]*eps1;

        KK1 = mvncdf(beta1',mu,sigma);
        beta_P1 = -norminv(KK1);
```

```
        alf1 = (beta_P1 - beta_s(h))/0.1;

        beta2 = -[M(2);M2(2)] - [M(3) M(4);M2(3) M2(4)]*eps2;

        KK2 = mvncdf(beta2',mu,sigma);
        beta_P2 = -norminv(KK2);

        alf2 = (beta_P2 - beta_s(h))/0.1;

        % Normalizing the calculated senstivity factors

        alphafy_s(h) = alf1/sqrt(alf1^2 + alf2^2);

        alphaF_s(h) = alf2/sqrt(alf1^2 + alf2^2);
        pair_s (h,:) = [M(1),M2(1)];
    end
end
beta_s = transpose(beta_s);
alphafy_s = transpose(alphafy_s);
alphaF_s = transpose(alphaF_s);

end
```

II.7.1.5 Matlab transcript of the parallel systems reliability analysis program at level 3

```
% This function is for the calculation of system reliability at level 3
% it takes the infromation from level 2 and the information from level 3 to
% calculate the system reliabilit of the whole structural system

function [beta_s,alphafy_s,alphaF_s] = Syslev3(Data2,crit3,delt)
h = 0
n1 = size(Data2)
alf = zeros(3,2)
LL = 0;
for i = 1:n1(1)
    M1 = Data2(i,:);% vector including the k th row of crit 1
    alf(1,:) = [M1(2),M1(3)]
    alf(2,:) = [M1(5),M1(6)]
    M2 = crit3{i}
    n2 = size(M2)
    for j = 1:n2(1)
        alf(3,:) = [M2(j,3),M2(j,4)]
        % calculating the correlation for the triple
        corln = zeros(3);
        for i1 = 1:3
            for j1 = 1:3
```

```
                if i1==j1
                   corln(i1,j1) = 1;
                else
                   corln(i1,j1)= alf(i1,:)*transpose(alf(j1,:));
                   if abs(corln(i1,j1)-1)>=1e-15
                   corln(i1,j1)=0.9999;
                   end
                end
            end
        end
        mu = [0 0 0];
        sigma = corln
        KK = mvncdf([-M1(1),-M1(4),-M2(j,2)],mu,sigma)
        LL = LL + 1
        beta_s(LL) = -norminv(KK)

        % calculating the equivalent sensitivity vectors
        eps1 = [0.1;0];
        eps2 = [0;0.1];

        beta1 = -[M1(1);M1(4);M2(j,2)] - alf*eps1

        KK1 = mvncdf(beta1',mu,sigma)
        beta_P1 = -norminv(KK1)

        alf1 = (beta_P1 - beta_s(LL))/0.1

        beta2 = -[M1(1);M1(4);M2(j,2)] - alf*eps2

        KK2 = mvncdf(beta2',mu,sigma)
        beta_P2 = -norminv(KK2)
        alf2 = (beta_P2 - beta_s(LL))/0.1
        %Normalizing the sensitivity vector
        alphafy_s(LL) = alf1/sqrt(alf1^2 + alf2^2)

        alphaF_s(LL) = alf2/sqrt(alf1^2 + alf2^2)

     end
end
beta_s = transpose(beta_s);
alphafy_s = transpose(alphafy_s);
alphaF_s = transpose(alphaF_s);

end
```

II.7.1.6 Matlab transcript of the reliability analysis program

```
%This is the system reliabiltiy module. it takes the influence factors and
%uses them to evaluate the reliability for each one of the safety margins
```

```
function [beta,alf_Fy,alf_F] = relanalysis(vec)

a = vec(1); %Resistanc coefficient
b = vec(2) %Load Effect coefficient
muF = 174;% Mean value of F
sigmaF = 60.9; %Standard deviation of F

muFy = 407;% Mean value of Fy
sigmaFy = 28.5;% Standard deviation of resistance

alpha = zeros(1,2);
F = 174;% Initial design point for load effect
yield = 407;% Initial design point for yield stress

beta = 20;
beta1 = 0;
res = beta - beta1;
n = 0;

while abs(res) >= 0.0001

    g = a*yield + b*F; %%%%%%%%%%%%%%%%5

    [muNf,sigmaNf] = gumbeleq(F,muF,sigmaF);

     uf = (F- muNf)/sigmaNf;

     [muNfy,sigmaNfy] = lognrmeq(yield,muFy,sigmaFy);

     ufy = (yield - muNfy)/sigmaNfy;

     Df =  b*sigmaNf; %%%%%%%%%%%%5

     Dfy = a*sigmaNfy;

     cof = (1/(Df^2 + Dfy^2))*(Df*uf + Dfy*ufy - g);

     n1 = cof*Df;
     n2 = cof*Dfy;

     beta = sqrt(n1^2 + n2^2);

     alpha(1) = Dfy/sqrt(Df^2 + Dfy^2); %resistance sensitivy factor
     alpha(2) = Df/sqrt(Df^2 + Dfy^2); %Force sensitivity factor

     F = muNf + sigmaNf*n1;
     yield = muNfy + sigmaNfy*n2;
```

```
        res = beta − beta1;
        beta1 = beta
        n = n + 1;
        if n>20
            beta = 100; % checking if algorithm diverges
            break
        end

end
alf_Fy = alpha(1);
alf_F = alpha(2);

if isnan(beta)
    beta = 101; %checking if algorithm returns a NaN (Not a Number)
end

if a<=0

    beta = 1003;% code to show if the resistance coefficient is negative
end

end
```

II.7.2 Transcript for the branch and bound system reliability method

The program for the branch and bound method uses the functions developed for the β-unzipping method. It gives the choice to the user to analyse different damaged states of the truss structure of Chapter 7 at either level 2 or level 3. The transcript for this program is shown below.

II.7.2.1 Matlab transcript of the branch and bound method

```
%This is the Program for the development of failure trees for the branch
%and bound method

%%%%%%%%%%%%%%%%%%%%%%%Structural Definition%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Defining the material properties and applied loads
F = [−1,−0.8];

E = 2*10^11;
A = [0.001961,0.000929,0.000188,0.000188,0.001226,0.00227,0.001049,0.000331,...
    0.000929,0.000778];
% Creating the geometrical Model

numberElements = 10;
numberNodes = 6;

elementNodes = [1 2;2 3;3 6;5 6;4 5;1 5;2 4;2 5;2 6;3 5];
```

```
nodeCoordinates = [0 0;3 0;6 0;0 4;3 4;6 4];

% Calculatin the orientaion of the elements
theta = zeros(0,numberElements);%theta, vector of element orientations

for i=1:numberElements
    a = elementNodes(i,:);
    b = nodeCoordinates(a(2),:);
    c = nodeCoordinates(a(1),:);
    delta = b' - c';
    theta(i) = atan(delta(2)/delta(1));
end

%member resistances with respect to compression

Rc = [1.22962 0.47776 0.00743 0.01307 0.62325 1.13555 0.31150 0.01767 0.22583...
    0.19073];
Rc = transpose(Rc);
% Member resistances with respect to tension
Rt = [1.91637 0.92928 0.18850 0.18850 1.22648 2.27043 1.0490 0.32987 0.92928...
    0.77833];
Rt = transpose(Rt);

xx = nodeCoordinates(:,1);
yy = nodeCoordinates(:,2);
% Forming the displacement vector, load vector and stiffness matrix

GDof = 2*numberNodes;
displacements = zeros(GDof,1);
force = zeros(GDof,1);

% applying the load on the structure

force(4)  = F(1)*1000;
force(6)  = F(2)*1000;

[stiffness] = formstiffness2Dtruss(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,xx,yy,E,A);

% Difnining boundry conditons

prescribedDof = [1 2 7 8];

% calculating the displacements stresses and intenal forces
displacements = solution(GDof,prescribedDof,stiffness,force);

[sigma,mforce] =  stresses2Dtruss(numberElements,elementNodes,...
    xx,yy,displacements,E,A);

 mforce = mforce/1000;
```

```
%%%%%%%%%%%%%%%%%%Calculation for the intact structure%%%%%%%%%%%%%%%%

% Calculating the beta values for the intact structure
result1 = zeros(numberElements,4);
mode = zeros(numberElements,1);
for i = 1:numberElements
     if mforce(i)<0
         [beta,alpha_fy,alpha_F] = relanalysis([Rc(i),mforce(i)]);
          mode(i) = 0; %Failure in compression

      else
         [beta,alpha_fy,alpha_F] = relanalysis([Rt(i),−mforce(i)]);
          mode(i) = 1; %Failure in Tension
      end
      result1(i,:) = [i,beta,alpha_fy,alpha_F];
end
matrix1 = [result1,mode];
% definig the number of elemnts that need to be removed
disp('The RIs for member the intact structure')
disp(matrix1)
beta = matrix1(:,2);
n = input('Enter the number of element to be removed:');
if n == 1
% Asking the user which member needs to be remove
    n1 = input('Enter the number of the element to be removed:');
    [members,betas,alphas,modes,aFy] = Damaged(A,E,mode(n1),numberElements,...
        numberNodes,elementNodes,...
       nodeCoordinates,Rc,Rt,n1,theta(n1));
% calculating the reliabiity index for the damaged state
    for i = 1:(numberElements−1)
        bb = mvncdf([−beta(n1),−betas(i)],[0 0],[1 0.999;0.999 1]);
        beta_s(i) = −norminv(bb);
    end

   disp(['RIs of the state where member ',num2str(n1),' is removed:'])
   disp([members,beta_s'])

elseif n==2
    n2 = input('Enter the vector of element numbers to be removed:');
    [members,betas,alphas,modes,aFy] = Damaged(A,E,mode(n2(1)),numberElements,...
        numberNodes,elementNodes,...
       nodeCoordinates,Rc,Rt,n2(1),theta(n2(1)));
    % applying the decrease in the number of elements
    L = 0;
    if n2(2)>n2(1)
        L = n2(2)−1
    else
        L = n2(2)
    end
```

```
    modes2 = [mode(n2(1)),modes(L)];
    thetas = [theta(n2(1)),theta(n2(2))];

    [members2,betas2,alphas2,modes3] = Damaged2(A,E,modes2,numberElements,...
        numberNodes,elementNodes,...
       nodeCoordinates,Rc,Rt,n2,thetas);

  % Calculating the reliability index for the damaged state
    L = 0;
    if n2(2)>n2(1)
        L = n2(2)-1
    else
        L = n2(2)
    end
    mu = [0 0 0];
    sigma = [1 0.9999 0.9999;0.9999 1 0.9999;0.9999 0.9999 1];

    for i = 1:(numberElements-2)
         bb = mvncdf([-beta(n2(1)),-betas(L),-betas2(i)],mu,sigma);
         beta_s(i) = -norminv(bb);
    end
    beta_s = transpose(beta_s);
    disp(['RIs for the damage state where members ',num2str(n2),' have failed are...
        :'])
    disp([members2,beta_s])

else

    disp('not able to remove more than 2 elements')
end
```

II.8 Transcript for the Example 5.11 in Reliability of Structures by Nowak

It was mentioned in Chapter 3 (Section 3.2.3.2) that the algorithm for the FORM method that is based on the Newton-Raphson recursive procedure was tested against an example in [37]. The Matlab transcript for this example is presented below.

```
% Example 5.11 in Reliability of Structures by A. S. Nowak and K.R. Collins
% to check the FORM method based on Newton-Raphson recursive procedure
m = 2000;
z = 100;
fy = 40;
beta1 = 0;
resBeta = 20;
g2 = 100;
b = 0;
```

```
sigma_z = 4;
sigma_fy = 4;
sigma_m = 200;
%place holders for the variables
a_z = 100;
a_m = 2000;
a_fy = 40;


while abs(resBeta) > 0.0001&& abs(g2) > 0.0001

        % first calculation of the performance function

        g1 = a_z*a_fy - a_m

        %calculating the equivalent normal parameters of the variables

        [muN_fy,sigmaN_fy] = lognrmeq(a_fy,fy,sigma_fy)

        [muN_m,sigmaN_m] = gumbeleq(a_m,m,sigma_m)

        b_z = (a_z - z)/sigma_z

        b_m = (a_m - muN_m)/sigmaN_m

        b_fy = (a_fy - muN_fy)/sigmaN_fy


        %calculating the partial derivative of the limit state fucntion by
        M1 = a_m;
        Z1 = a_z;
        Fy1 = a_fy;

        percentM = 0.01;
        percentZ = 0.004;
        percentFy = 0.01
        M2 = M1 + (percentM*M1)

        Z2 = Z1 + (percentZ*Z1)

        Fy2 = Fy1 + (percentFy*Fy1)

        G2(1) = Z1*Fy1 - M2
        G2(2) = Z2*Fy1 - M1
        G2(3) = Z1*Fy2 - M1

        DG_M = (G2(1) - g1)/(M2- M1)
        DG_Z = (G2(2) - g1)/(Z2- Z1)
        DG_Fy = (G2(3) - g1)/(Fy2- Fy1)
```

```
        %calculating the partial derivative in the equivalent normal space

        DGN_M = DG_M*sigmaN_m
        DGN_Z = DG_Z*sigma_z
        DGN_Fy = DG_Fy*sigmaN_fy

        %calculating the coordinates of the new design point

        k = (1/(DGN_M^2+DGN_Z^2+DGN_Fy^2))*(DGN_M*b_m + DGN_Z*b_z ...
            + DGN_Fy*b_fy - g1)

        m_new = k*DGN_M
        z_new = k*DGN_Z
        fy_new = k*DGN_Fy

        %calculating the beta value of the member & checking the beta
        %convergence

        beta = sqrt(m_new^2 + z_new^2 + fy_new^2)

        resBeta = beta- beta1

        beta1 = beta;

        %calculating the coordinates of the new iteration point

        a_m = muN_m+ sigmaN_m*m_new
        a_z = z + sigma_z*z_new
        a_fy = muN_fy + sigmaN_fy*fy_new

        g2 = a_z*a_fy - a_m



        b = b + 1
        %if b > 20
            %break
        %end
end
```

Appendix III

III.9 System reliability analysis for non-normal random variables

The detailed calculations for the system reliability analysis of the case where both of the random variables are non-normal is presented in Sections III.9.1 to III.9.4.

III.9.1 System reliability analysis at level zero:

The reliability indices of the components of the truss are shown in Table 1. These calculations were performed in Chapter 5.

Table 1: Reliability indices of the truss elements at level zero

Member	β	P_f
1	2.4850	0.00648
2	2.4593	0.00696
3	7.0260	1.06E-12
4	7.2222	2.56E-13
5	2.4852	0.00647
6	2.6263	0.00431
7	2.3855	0.00853
8	7.0296	1.03E-12
9	6.9998	1.281E-12
10	2.3944	0.00832
Minimum RI	2.3855	0.00853

The system reliability at this level is equal to the reliability of the member with the lowest reliability index which is member 7.

$$\beta_s^0 = \min \beta_i = \beta_7 = 2.3855$$

III.9.2 System reliability analysis at level one

At level one the reliability of the system is calculated with the assumption of $\Delta\beta = 1$. The assumption will yield the interval of $[2.3855, 3.3855]$. The series system that is obtained according to this interval is exactly the same as the one obtained for the case where all of the basic random variables were normal. Therefore, components 1,2,5,6,7, and 10 have reliability indices that are within the aforementioned unzipping interval.

For the evaluation of the system reliability at this level, the correlation coefficients between the components of the series system has to be obtained. The first step is the calculation of the sensitivity factors for each of the components. The sensitivity calculation is performed through the iterative procedure of FORM method using Equation 3.15 which is shown below.

$$\alpha_i = \frac{\left(\frac{\partial g}{\partial X_i}\right)^* \sigma_{X_i}^N}{\sqrt{\sum\limits_{i=1}^{n} \left(\frac{\partial g}{\partial X_i}\right)^{2*}}}$$

The sensitivity factors that are calculated for the members are shown in Table 2.

Table 2: Sensitivity factors at level 1

Member	a	b	α_R	α_E
1	1.2293	1.2767	0.19728	-0.98035
2	0.4778	0.5005	0.19715	-0.98037
5	1.2260	1.2733	0.19728	-0.98035
6	1.1352	1.1222	0.19812	-0.98018
7	1.0490	1.1278	0.19681	-0.98044
10	0.7783	0.8342	0.19684	-0.98043

The correlation matrix is calculated as shown below. This matrix is a 6×6 matrix corresponding to the elements with reliability values within the β-unzipping interval.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

Using the Ditlevsen bounds for the series system, the failure probability of the system at level one is within the interval that is presented below.

$$8.5270 \times 10^{-3} \leq P_f^1 \leq 4.0387 \times 10^{-2}$$

Form the correlation coefficient matrix, it is obvious that the safety margins of the elements of the series system are fully correlated; hence the lower bound of Ditlevsen interval provides a good estimate of the system reliability at level 1.

$$P_{f_{sys}}^1 = 8.5270 \times 10^{-3} \rightarrow \beta_s^1 = 2.3855$$

III.9.3 System Reliability analysis at level two:

For the system reliability at level two the unzipping is based on a value of $\Delta\beta_1 = 0.075$ for the system reliability at level one. This helps reduce the members that have to be removed from the structure for system reliability at level 2. With this value, the interval that is used for the unzipping is equal to $[2.3855, 2.4605]$. Form the reliability values shown in Table 1, it is clear that components 7, 10 and 2 have reliability values within the interval.

Removing member 7: Member 7 is the first member to be removed from the structure. The influence factors for the damaged state of the structure where member 7 is in its failure state were calculated in Section 7.2.1.3. Also the applicability and prevalence of safety margins are the same, and, accordingly, the reliability indices for the remaining members can be computed. This is shown in Table 3.

Table 3: Safety margins for failure in compression($G^-_{i|7}$) and tension ($G^+_{i|7}$)-member 7 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^-_{1\|7} = 1.2296f_y + (-0.6000F - 0.6294R^+_7)$	2.5817	Compression
2	$G^-_{2\|7} = 0.4778f_y + (-0.6311F + 0.1215R^+_7)$	2.4344	Compression
3	$G^+_{3\|7} = 0.1885f_y - (-0.0415F + 0.1620R^+_7)$	N/A	Tension
4	$G^-_{4\|7} = 0.0131f_y + (-0.0311F + 0.1215R^+_7)$	7.5037	Compression
5	$G^+_{5\|7} = 1.2265f_y - (+1.9500F - 0.6294R^+_7)$	2.4411	Tension
6	$G^-_{6\|7} = 1.1356f_y + (-2.2500F + 1.0490R^+_7)$	2.4974	Compression
8	$G^+_{8\|7} = 0.3299f_y - (+0.9585F - 0.6772R^+_7)$	2.7211	Tension
9	$G^-_{9\|7} = 0.2258f_y + (+0.0519F - 0.2025R^+_7)$	N/A	Compression
10	$G^+_{10\|7} = 0.7783f_y - (+1.0519F - 0.2025R^+_7)$	2.3833	Tension

From the reliability indices shown in Table 3, it can be seen that member 10 has the smallest reliability index followed by members 5 and 2. It should also be noted that the reliability indices of members 3 and 10 could not be calculated due to the divergence of the algorithm. However, in section 7.2.1.3, it was observed that these two components possess high reliability index values, and don't play an important role in the unzipping process. Thus these two failure components are ignored for the damaged state of member 7 failure.

With the assumption of $\Delta\beta_2 = 0.1$, the unzipping interval is equal to $[2.3833, 2.4833]$. Failure elements 10,5, and 2 have reliability indices within the interval. The parallel systems that are formed here are the same the ones shown in Figure 7.4.

The sensitivity factors as well as the correlation coefficients of each one of these elements are shown in Table 4. The sensitivity vector for element 7 is $[0.1968, -0.9804]$.

Table 4: Sensitivity factors and correlation coefficients-member 7 in failure state

Member	α_F	α_{f_y}	$\rho_{i\|7}$
2	0.19707	-0.98039	0.9999
5	0.19711	-0.98038	0.9999
10	0.19684	-0.98044	0.9999

The reliability indices and probabilities of failure of each pair can be calculated as shown in Table 5.

Table 5: RI and P_f values of the failure element pairs-member 7 in failure state

Member	β_7	$\beta_{i\|7}$	$P_{f_{pair}}$	β_{pair}
2	2.3855	2.4344	0.0075	2.4344
5	2.3855	2.4411	0.0073	2.4411
10	2.3855	2.3833	0.0084	2.3901

The equivalent safety margins for each pair are shown below.

Table 6: Equivalent safety margins-member 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7	0.1971	-0.9804	$G^e_{2,7} = 0.1971f_y - 0.9804F + 2.4344$
5 & 7	0.1971	-0.9804	$G^e_{5,7} = 0.1971f_y - 0.9804F + 2.4411$
10 & 7	0.1971	-0.9804	$G^e_{10,7} = 0.1971f_y - 0.9804F + 2.3901$

Removing member 2: Next damaged state that needs to be considered at level 2 is the damaged state where member 2 has failed. This damaged state was already investigated for the case of normal input random variables. For the case of non-normal random variables the applicable limit states and failure modes need to be evaluated. The results are shown in Table 7.

Table 7: Safety margins for failure in compression($G^-_{i|2}$) and tension ($G^+_{i|2}$)-member 2 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^-_{i\|2} = 1.2296f_y + (-1.5174F + 0.2290f_y)$	2.4694	Compression
3	$G^+_{i\|2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$	2.6701	Tension
4	$G^+_{i\|2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$	2.8790	Tension
5	$G^+_{i\|2} = 1.2265f_y - (+1.0326F + 0.2290f_y)$	2.4828	Tension
6	$G^-_{i\|2} = 1.1356f_y + (-0.7210F - 0.3817f_y)$	2.7070	Compression
7	$G^+_{i\|2} = 1.0490f_y - (+1.5290F - 0.3817f_y)$	2.3933	Tension
8	$G^+_{i\|2} = 0.3299f_y - (+0.5768F - 0.3317f_y)$	2.9719	Tension
9	$G^-_{i\|2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	2.6430	Compression
10	$G^-_{i\|2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	N/A	Compression

Member 7 has the smallest reliability index which is equal to 2.3933. As a result, the unzipping interval is equal to $[2.3933, 2.4933]$. Members 7, 5, and 1 are the failure elements with reliability index values within the unzipping interval. The pair of elements 7 and 2 was already investigated for the damaged state where member 7 had failed, but the sequence here is different and it needs to be considered as well. However, the pairs of elements 2 & 1, 2 & 5, and 2 & 7 need to be assessed. It should be noted that these pairs are the same as the ones shown in Figure 7.8.

The sensitivity factors and correlation coefficient between the safety margin of member 2 with that of elements 1, 5, and 7 are given in Table 8.

Table 8: Sensitivity factors and correlation coefficients-member 2 in failure state

Member	α_F	α_{f_y}	$\rho_{i\|2}$
7	0.19688	-0.98043	0.9999
1	0.19725	-0.98035	0.9999
5	0.19732	-0.98034	0.9999

The sensitivity vector of member 2 is $[0.19715, -0.98037]$ which is used to calculate the correlation between the safety margins of elements 1, 5, and 7 with the safety margin of failure element 2.

The reliability indices and failure probabilities are calculated for both of pairs of failure elements. These values are shown in Table 9.

Table 9: RI and P_f values of the failure element pairs-member 2 in failure state

Member	β_2	$\beta_{i\|2}$	$P_{f_{pair}}$	β_{pair}
7	2.4593	2.3933	0.0070	2.4593
1	2.4593	2.4694	0.0067	2.4714
5	2.4593	2.4828	0.0065	2.4831

The equivalent safety margins are shown in Table 10.

Table 10: Equivalent safety margins-member 2 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7	0.1971	-0.9804	$G^e_{2,7} = 0.1971f_y - 0.9804F + 2.4593$
2 & 1	0.1971	-0.9804	$G^e_{2,1} = 0.1971f_y - 0.9804F + 2.4714$
2 & 5	0.1971	-0.9804	$G^e_{5,2} = 0.1971f_y - 0.9804F + 2.4831$

Removing member 10: The last damaged state that needs to be considered in level 2 is the damaged state where member 10 has failed. The safety margins that should be investigated for this damaged state are shown in Table 11.

Table 11: Safety margins for failure in compression($G^-_{i|10}$) and tension ($G^+_{i|10}$)-member 10 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^-_{1\|10} = 1.2296f_y + (-1.5174F + 0.2238f_y)$	2.4593	Compression
2	N/A	N/A	NA
3	$G^+_{3\|10} = 0.1885f_y - (+0.8000F - 0.6226f_y)$	2.6198	Tension
4	$G^+_{4\|10} = 0.1885f_y - (+0.6000F - 0.4670f_y)$	2.8323	Tension
5	$G^+_{5\|10} = 1.2265f_y - (+1.0326F + 0.2238f_y)$	2.4975	Tension
6	$G^-_{6\|10} = 1.1356f_y + (-0.7210F - 0.3731f_y)$	2.7396	Compression
7	$G^+_{7\|10} = 1.0490f_y - (+1.5290F - 0.3731f_y)$	2.3763	Tension
8	$G^+_{8\|10} = 0.3299f_y - (+0.5768F - 0.3242f_y)$	2.9387	Tension
9	$G^-_{9\|10} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	2.5924	Compression

Form the reliability indices shown in Table 11, the minimum reliability index belongs to member 7. With the assumption of $\Delta\beta_2 = 0.1$ the unzipping interval is $[2.3763, 2.4763]$. Only members 7 and 1 have reliability values within the interval.

The pair of elements 10 and 7 was already investigated. Nevertheless, the sequence is different and this parallel pair need to be considered. Therefore, the pair of elements 10 & 1 and 10 & 7 should be investigated.

The vector of sensitivity factors for member 10 is $[0.19684, -0.98043]$. The sensitivity factors and correlation coefficients with the safety margin of member 10 are shown in Table 12.

Table 12: Sensitivity factors and correlation coefficients-member 10 in failure state

Member	α_F	α_{f_y}	$\rho_{i\|2}$
7	0.98044	-0.19680	0.9999
1	0.19728	-0.98035	0.9999

The reliability indices for the pairs of elements 10 & 1 and 10 & 7 are presented in Table 13.

Table 13: RI and P_f values of the failure element pairs-member 10 in failure state

Member	β_{10}	$\beta_{i\|10}$	$P_{f_{pair}}$	β_{pair}
7	2.3944	2.3763	0.0083	2.3951
1	2.3944	2.4850	0.0065	2.4850

The equivalent safety margin for the pair of elements 1 and 10 is shown in Table 14.

Table 14: Equivalent safety margins-member 10 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
10 & 7	0.1969	-0.9804	$G^e_{10,1} = 0.1969f_y - 0.9804F + 2.3951$
10 & 1	0.1971	-0.9804	$G^e_{10,1} = 0.1971f_y - 0.9804F + 2.4850$

The structural system can be modelled precisely the same way it was modelled for the case of normally distributed basic random variables. The model for the system is shown in Figure 2 below.

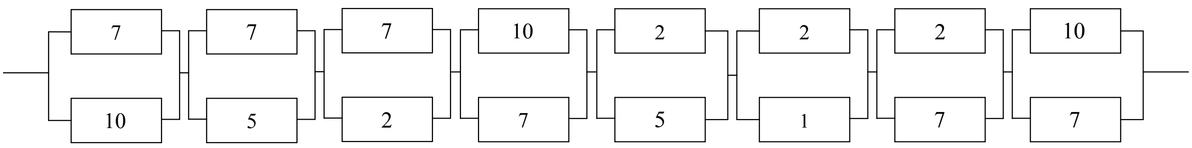

Figure 2: Ten-bar truss structure modelled as parallel-series system at level 2

An outline of the equivalent safety margins for the parallel pairs of elements is provided through Table 15.

Table 15: Critical parallel pairs at level 2 and their equivalent safety margins and reliability indices

Parallel pair	Equivalent limit state	Reliability index
$7 \rightarrow 2$	$G^e_{2,7} = 0.1971 f_y - 0.9804F + 2.4344$	2.4344
$7 \rightarrow 5$	$G^e_{5,7} = 0.1971 f_y - 0.9804F + 2.4411$	2.4411
$7 \rightarrow 10$	$G^e_{10,7} = 0.1971 f_y - 0.9804F + 2.3901$	2.3901
$2 \rightarrow 7$	$G^e_{2,7} = 0.1971 f_y - 0.9804F + 2.4593$	2.4593
$2 \rightarrow 1$	$G^e_{2,1} = 0.1971 f_y - 0.9804F + 2.4714$	2.4714
$2 \rightarrow 5$	$G^e_{5,2} = 0.1971 f_y - 0.9804F + 2.4831$	2.4831
$10 \rightarrow 7$	$G^e_{10,1} = 0.1969 f_y - 0.9804F + 2.3951$	2.3951
$10 \rightarrow 1$	$G^e_{10,1} = 0.1971 f_y - 0.9804F + 2.4850$	2.4850

The correlation coefficient matrix is calculated as below.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

Applying the Ditlevsen bounds for the system will lead to the following interval for the system probability of failure.

$$8.4219 \times 10^{-3} \leq P_f^1 \leq 4.2162 \times 10^{-2}$$

Due to the high correlation between the equivalent safety margins of the parallel pairs, the lower Ditlevsen bound will provide a good estimate for the system reliability.

$$P^2_{f_{sys}} = 8.4219 \times 10^{-3} \rightarrow \beta^2_s = 2.3901$$

III.9.4 System reliability Analysis at level Three

System reliability analysis at level 3 can be performed as was done in Section 7.2.1.4. The damaged states identified in Section 7.2.2.3 for level two need to be investigated to further assess the system reliability at level 3. These damage states were already investigated in Section 7.2.1.4; therefore, the data provided in that section can be used for the case of non-normal random variables.

In Table 16 the critical pairs of failure elements and their corresponding failure modes as well as the reliability indices are shown.

Table 16: Pairs of failure elements at level two

Pair of i & j	Failure mode of i	Failure mode of j	Reliability index
7 & 2	Tension	Compression	2.4344
7 & 5	Tension	Tension	2.4411
7 & 10	Tension	Tension	2.3901
2 & 1	Compression	Compression	2.4714
2 & 5	Compression	Tension	2.4831
10 & 7	Tension	Tension	2.3951
10 & 1	Tension	Compression	2.4850

For this case, $\Delta\beta_3$ is chosen to be 0.1 which is equal to the value chosen for the unzipping procedure at level 2.

Removing members 7 and 2: The first pair to be investigated is the pair of elements 7 and 2. The governing failure modes and safety margins are shown in Table 17.

Table 17: Safety margins for failure in compression ($G^-_{i|7,2}$) and tension ($G^+_{i|7,2}$)-members 7 & 2 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^-_{1\|7,2} = 1.2296f_y + (-0.6000F - 0.6294f_y)$	2.5817	Compression
3	$G^+_{3\|7,2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$	2.6701	Tension
4	$G^+_{4\|7,2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$	2.8790	Tension
5	$G^+_{5\|7,2} = 1.2265f_y - (+1.9500F - 0.6294f_y)$	2.4411	Tension
6	$G^-_{6\|7,2} = 1.1356f_y + (-2.2500F + 1.0490f_y)$	2.4974	Compression
8	$G^+_{8\|7,2} = 0.3299f_y - (+1.8000F + 1.4763f_y)$	2.5904	Tension
9	$G^-_{9\|7,2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	2.6430	Compression
10	$G^-_{10\|7,2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	8.0624	Compression

According to Table 17, member 5 has the smallest reliability index. This will yield the unzipping interval of $[2.4411, 2.5411]$. The parallel triples that will be formed are the triples of members 7-2-5 and 7-2-6 since only the reliability indices of members 6 and 5 are within the interval.

Table 18 shows the sensitivity factors and correlation coefficients in the damaged state of members 2 and 7 as failed elements.

Table 18: Sensitivity factors and correlation coefficients-members 7 & 2 in failure state

Member	α_{f_y}	α_F	$\rho_{i,7}$	$\rho_{i,2}$
5	0.19711	-0.98038	0.9999	0.9999
6	0.19740	-0.98032	0.9999	0.9999

The correlation coefficient between the safety margins of members 7 and 2 was computed in Section 7.2.1.4 as being equal to 0.9999.

The reliability indices and failure probabilities are calculated for each one of the parallel triples. These values are presented in Table 19.

Table 19: RI and P_f values for the triples of failure elements-members 7 & 2 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
7 & 2 & 5	2.3855	2.4344	2.4411	0.0073	2.4440
7 & 2 & 6	2.3855	2.4344	2.4974	0.0063	2.4974

The equivalent safety margins for the parallel triples are obtained as presented in Table 20.

Table 20: Equivalent safety margins-members 7 & 2 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
5 & 7 & 2	0.1980	-0.9802	$G^e_{7\ \&\ 2\ \&\ 5} = 0.1980f_y - 0.9802F + 2.4440$
6 & 7 & 2	0.1971	-0.9804	$G^e_{7\ \&\ 2\ \&\ 6} = 0.1971f_y - 0.9804F + 2.4974$

Another failure sequence that corresponds to this damaged state is the the sequence of element 2 and 7. It will form the triple parallels of element 2 & 7 & 5 and 2 & 7 & 6. The sensitivity factors and correlation coefficients are the same. However, the failure probabilities as well as the equivalent safety margins might be different which are given in Tables 21 and 22 below.

Table 21: RI and P_f values for the triples of failure elements-members 2 & 7 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
2 & 7 & 5	2.4593	2.2.3933	2.4411	0.0069	2.4600
2 & 7 & 6	2.4593	2.2.3933	2.4974	0.0063	2.4974

Table 22: Equivalent safety margins-members 2 & 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 7 & 5	0.1971	-0.9804	$G^e_{2\ \&\ 7\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4600$
2 & 7 & 6	0.1974	-0.9803	$G^e_{2\ \&\ 7\ \&\ 6} = 0.1974f_y - 0.9803F + 2.4974$

Removing members 7 and 5: The next damaged state of the structure is the damaged state where members 7 and 5 have failed. However, as mentioned in Section 7.2.1.4 this will cause the collapse of the whole system; hence this damaged state cannot be considered at level 3.

Removing members 7 and 10: The next pair of failed elements is the pair of elements 10 and 7. The reliability indices for this damaged state are shown along with the safety margins of the remaining members.

Table 23: Safety margins for failure in compression($G^{-}_{i|10,7}$) and tension ($G^{+}_{i|10,7}$)-members 7 & 10 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^{-}_{1\|10,7} = 1.2296f_y + (-0.6000F - 0.6294f_y)$	2.5817	Compression
2	N/A	8.2670	Tens/Comp
3	$G^{+}_{3\|10,7} = 0.1885f_y - (+0.8000F - 0.6226f_y)$	2.6200	Tension
4	$G^{+}_{4\|10,7} = 0.1885f_y - (+0.6000F - 0.4670f_y)$	2.8322	Tension
5	$G^{+}_{5\|10,7} = 1.2265f_y - (+1.9500F - 0.6294f_y)$	2.4455	tension
6	$G^{-}_{6\|10,7} = 1.1356f_y + (-2.2500F + 1.0490f_y)$	2.4974	Compression
8	$G^{+}_{8\|10,7} = 0.3299f_y - (+1.8000F + 1.4618f_y)$	2.5677	Tension
9	$G^{-}_{9\|10,7} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	2.5925	Compression

With the value of $\Delta\beta_3$ being equal to 0.1, the interval will be $[2.4455, 2.5455]$. Again, members 5 and 6 are the components with reliability indices within the interval. The sensitivity factors and correlation coefficients for this damaged state are presented in Table 24.

Table 24: Sensitivity factors and correlation coefficients-members 7 & 10 in failure state

Member	α_{f_y}	α_F	$\rho_{i,7}$	$\rho_{i,10}$
5	0.19713	-0.98038	0.9999	0.9999
6	0.19740	-0.98032	0.9999	0.9999

The correlation coefficient between the safety margins of member 7 and 10 was computed as being equal to 0.9999.

The results of the reliability calculation for the two parallel triples are presented in Table 25.

Table 25: RI and P_f values for the triples of failure elements-members 7 & 10 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{triple}}$	β_{triple}
7 & 10 & 5	2.3855	2.3833	2.4455	0.0072	2.4455
7 & 10 & 6	2.3855	2.3833	2.4974	0.0063	2.4974

The equivalent safety margins are shown in Table 26.

Table 26: Equivalent safety margins for the damaged state-members 7 & 10 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
7 & 10 & 5	0.1971	-0.9804	$G^e_{7\ \&\ 10\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4455$
7 & 10 & 6	0.1971	-0.9804	$G^e_{7\ \&\ 10\ \&\ 6} = 0.1971f_y - 0.9804F + 2.4974$

Another failure sequence that leads to this physical damaged state is the failure sequence of elements 10 and 7. The results for failure probability and reliability index of this failure sequences are given in Table 27. The equivalent safety margins are given in Table 28.

Table 27: RI and P_f values for the triples of failure elements-members 10 & 7 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
10 & 7 & 5	2.3944	2.3763	2.4455	0.0072	2.4455
7 & 10 & 6	2.3944	2.3763	2.4974	0.0063	2.4974

Table 28: Equivalent safety margins-members 10 & 7 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
10 & 7 & 5	0.1971	-0.9804	$G^e_{10\ \&\ 7\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4455$
10 & 7 & 6	0.1974	-0.9803	$G^e_{10\ \&\ 7\ \&\ 6} = 0.1974f_y - 0.9803F + 2.4974$

Removing members 2 and 1: The damaged state where members 2 and 1 have failed is the next damaged state to be considered. The reliability indices and safety margins are shown for this damaged state in Table 29.

Table 29: Safety margins for failure in compression($G^-_{i|1,2}$) and tension ($G^+_{i|1,2}$)-members 2 & 1 in failure state

Member	Safety Margin	Reliability index	Failure mode
3	$G^+_{3\|1,2} = 0.1885f_y - (+0.8000F - 0.6371f_y)$	2.6701	Tension
4	$G^+_{4\|1,2} = 0.1885f_y - (+0.6000F - 0.4778f_y)$	2.8790	Tension
5	$G^+_{5\|1,2} = 1.2265f_y - (+2.5500F - 1.2296f_y)$	2.2359	Tension
6	$G^-_{6\|1,2} = 1.1356f_y + (-3.2500F + 2.0493f_y)$	2.5236	Compression
7	$G^-_{7\|1,2} = 0.3115f_y + (-1.0000F + 2.0493f_y)$	5.2178	Compression
8	$G^+_{8\|1,2} = 0.3299f_y - (+2.6000F - 2.2766f_y)$	2.5878	Tension
9	$G^-_{9\|1,2} = 0.2258f_y + (-1.0000F + 0.7963f_y)$	2.6430	Compression
10	$G^-_{10\|1,2} = 0.1907f_y + (+0.0000F + 0.7963f_y)$	8.0624	Compression

Using the minimum reliability index and $\Delta\beta_3 = 0.1$, the unzipping interval is obtained as $[2.2359, 2.3359]$. Only member 5 has a reliability value that is within the interval. Table 30 shows the correlation coefficients and sensitivity factor for member 5 in the damaged state.

Table 30: Sensitivity factors and correlation coefficients-members 2 & 1 in failure state

Member	α_{f_y}	α_F	$\rho_{i,2}$	$\rho_{i,1}$
5	0.19630	-0.98054	0.9999	0.9999

The reliability index is calculated as shown in Table 31.

Table 31: RI and P_f values for the triples of failure elements-members 2 & 1 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
2 & 1 & 5	2.4593	2.4694	2.2359	0.0067	2.4714

The equivalent safety margin for the triple of elements 1,2 and 5 is presented in Table 32.

Table 32: Equivalent safety margins-members 2 & 1 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 1 & 5	0.1977	-0.9803	$G^e_{2\ \&\ 1\ \&\ 5} = 0.1977 f_y - 0.9803F + 2.4714$

Removing members 2 and 5: The damaged state where members 2 & 5 have failed should be considered as the fifth damaged state of the structure for the system reliability analysis at level 3. The dominant failure modes and reliability indices are shown in Table 33.

Table 33: Safety margins for failure in compression($G^-_{i|2,5}$) and tension ($G^+_{i|2,5}$)-members 2 & 5 in failure state

Member	Safety Margin	Reliability index	Failure mode
1	$G^-_{1\|2,5} = 1.2296 f_y + (-2.5500F + 1.2265 f_y)$	2.4748	Compression
3	$G^+_{3\|2,5} = 0.1885 f_y - (+0.8000F - 0.6371 f_y)$	2.6699	Tension
4	$G^+_{4\|2,5} = 0.1885 f_y - (+0.6000F - 0.4778 f_y)$	2.8790	Tension
6	$G^+_{6\|2,5} = 2.2704 f_y - (+1.0000F - 2.0442 f_y)$	7.4909	Tension
7	$G^+_{7\|2,5} = 1.0490 f_y - (+3.2500F - 2.0442 f_y)$	2.4411	Tension
8	$G^-_{8\|2,5} = 0.0177 f_y + (-0.8000F + 0.9982 f_y)$	3.2670	Compression
9	$G^-_{9\|2,5} = 0.2258 f_y + (-1.0000F + 0.7963 f_y)$	2.6429	Compression
10	$G^-_{10\|2,5} = 0.1907 f_y + (+0.0000F + 0.7963 f_y)$	8.0619	Compression

Based on the minimum reliability index ($\beta_7 = 2.4411$), the unzipping interval of $[2.4411, 2.5411]$ is obtained.
Members 7 and 1 are the only members with reliability values within the interval this will yield the triples of 2 & 5 & 7 and 2 & 5 & 1. Although, these two parallel triples were already investigated, They need to be considered here since they are obtained through a different sequence.

The sensitivity factors and correlation coefficients for this parallel system are presented in Table 34.

Table 34: Sensitivity factors and correlation coefficients-members 2 & 5 in failure state

Member	α_{f_y}	α_F	$\rho_{i,2}$	$\rho_{i,5}$
1	0.19728	-0.98035	0.9999	0.9999
7	0.19711	-0.98038	0.9999	0.9999

The reliability indices of the elements and the parallel triple are given in Table 35.

Table 35: RI and P_f values for the triples of failure elements-member 2 & 5 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{pair}}$	β_{triple}
2 & 5 & 1	2.4593	2.4828	2.4748	0.0065	2.4855
2 & 5 & 7	2.4593	2.4828	2.4411	0.0065	2.4831

The equivalent safety margins are shown in Table 36.

Table 36: Equivalent safety margins-members 2 & 5 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
2 & 5 & 1	0.1973	-0.9803	$G^e_{2\ \&\ 5\ \&\ 1} = 0.1973f_y - 0.9803F + 2.4855$
2 & 5 & 7	0.1973	-0.9803	$G^e_{2\ \&\ 5\ \&\ 7} = 0.1973f_y - 0.9803F + 2.4831$

Removing members 10 and 1: The last damaged state that should be investigated at level 3 is the state where members 1 and 10 have failed. The governing safety margins together with the reliability indices of the remaining elements are given in Table 37.

Table 37: Safety margins for failure in compression ($G^-_{i|10,1}$) and tension ($G^+_{i|10,1}$)-members 10& 1 in failure state

Member	Safety Margin	Reliability index	Failure mode
2	$G^+_{2\|1,10} = 0.9293f_y - (+0.0000F - 0.4670f_y)$	N/A	Compression
3	$G^+_{3\|1,10} = 0.1885f_y - (+0.8000F - 0.6226f_y)$	2.6198	Tension
4	$G^+_{4\|1,10} = 0.1885f_y - (+0.6000F - 0.4670f_y)$	2.8323	Tension
5	$G^+_{5\|1,10} = 1.2265f_y - (+2.5500F - 1.2296f_y)$	2.4748	Tension
6	$G^+_{6\|1,10} = 2.2704f_y - (-3.2500F + 1.0493f_y)$	2.5236	Tension
7	$G^-_{7\|1,10} = 0.3115f_y + (-1.0000F + 2.0493f_y)$	5.2178	Compression
8	$G^+_{8\|1,10} = 0.3299f_y - (+2.6000F - 2.2621f_y)$	2.5721	Compression
9	$G^-_{9\|1,10} = 0.2258f_y + (-1.0000F + 0.7783f_y)$	2.5924	Compression

Based on the minimum reliability index and $\Delta\beta_3$, the β-unzipping interval will be equal to $[2.4748, 2.5748]$. According to the interval the parallel triples of elements 1 & 10 & 5 and 1 & 10 & 6 need to be investigated.

The sensitivity factors for the critical elements 5 and 6 are given in Table 38.

Table 38: Sensitivity factors and correlation coefficients-members 10 & 1 in failure state

Member	α_{f_y}	α_F	$\rho_{i,1}$	$\rho_{i,10}$
5	0.19728	-0.98035	0.9999	0.9999
6	0.19755	-0.98029	0.9999	0.9999

Using the reliability indices and correlation coefficients, it is possible to calculate the system reliability for each parallel triple.

Table 39: RI and P_f values for the triples of failure elements-members 10 & 1 in failure state

Member	β_i	$\beta_{i\|j}$	$\beta_{k\|i,j}$	$P_{f_{triple}}$	β_{triple}
10 & 1 & 5	2.3944	2.4850	2.4748	0.0064	2.4870
10 & 1 & 6	2.3944	2.4850	2.5236	0.0058	2.5236

The equivalent safety margins are calculated as presented in Table 40.

Table 40: Equivalent safety margins-member 10 & 1 in failure state

Parallel system	$\alpha^e_{f_y}$	α^e_F	Safety Margin
10 & 1 & 5	0.1969	-0.9804	$G^e_{10\ \&\ 1\ \&\ 5} = 0.1969f_y - 0.9804F + 2.4870$
10 & 1 & 6	0.1980	-0.9802	$G^e_{10\ \&\ 1\ \&\ 6} = 0.1980f_y - 0.9802F + 2.5236$

Table 41 shows an outline of the equivalent safety margins and failure probabilities.

Table 41: Critical parallel triples at level 3 and their equivalent safety margins and reliability indices

Parallel triple	Equivalent limit state	Failure probability	RI
$7 \rightarrow 2 \rightarrow 5$	$G^e_{7\ \&\ 2\ \&\ 5} = 0.1980f_y - 0.9802F + 2.4440$	0.0073	2.4440
$7 \rightarrow 2 \rightarrow 6$	$G^e_{7\ \&\ 2\ \&\ 6} = 0.1971f_y - 0.9804F + 2.4974$	0.0063	2.4974
$2 \rightarrow 7 \rightarrow 5$	$G^e_{2\ \&\ 7\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4600$	0.0069	2.4600
$2 \rightarrow 7 \rightarrow 6$	$G^e_{2\ \&\ 7\ \&\ 6} = 0.1974f_y - 0.9803F + 2.4974$	0.0063	2.4974
$7 \rightarrow 10 \rightarrow 5$	$G^e_{7\ \&\ 10\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4455$	0.0072	2.4455
$7 \rightarrow 10 \rightarrow 6$	$G^e_{7\ \&\ 10\ \&\ 6} = 0.1971f_y - 0.9804F + 2.4974$	0.0063	2.4974
$10 \rightarrow 7 \rightarrow 5$	$G^e_{10\ \&\ 7\ \&\ 5} = 0.1971f_y - 0.9804F + 2.4455$	0.0072	2.4455
$10 \rightarrow 7 \rightarrow 6$	$G^e_{10\ \&\ 7\ \&\ 6} = 0.1974f_y - 0.9803F + 2.4974$	0.0063	2.4974
$2 \rightarrow 1 \rightarrow 5$	$G^e_{1\ \&\ 2\ \&\ 5} = 0.1977f_y - 0.9803F + 2.4714$	0.0067	2.4714
$2 \rightarrow 5 \rightarrow 1$	$G^e_{2\ \&\ 5\ \&\ 1} = 0.1973f_y - 0.9803F + 2.4855$	0.0065	2.4855
$2 \rightarrow 5 \rightarrow 7$	$G^e_{2\ \&\ 5\ \&\ 7} = 0.1973f_y - 0.9803F + 2.4831$	0.0065	2.4831
$10 \rightarrow 1 \rightarrow 5$	$G^e_{10\ \&\ 1\ \&\ 5} = 0.1969f_y - 0.9804F + 2.4870$	0.0064	2.4870
$10 \rightarrow 1 \rightarrow 6$	$G^e_{10\ \&\ 1\ \&\ 6} = 0.1980f_y - 0.9802F + 2.5236$	0.0058	2.5236

The parallel-series model that can be used for system reliability calculation is a model composed of 13 parallel systems which form a series system. This is shown in Figure 3.

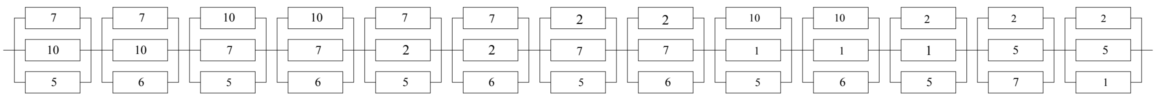

Figure 3: Ten-bar truss structure modelled as a parallel-series system at level 3

The correlation coefficient matrix for the parallel systems is as below. As is seen, the matrix is 13×13 which corresponds to the the considered number of failure sequences.

$$\rho = \begin{bmatrix} 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 & 0.99 \\ 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 0.99 & 1 \end{bmatrix}$$

The Ditlevsen bounds for the series system composed of parallel triples of failure elements is shown below.

$$7.2627 \times 10^{-3} \leq P_f^3 \leq 8.2351 \times 10^{-2}$$

Due to the high correlation between the safety margins of the parallel triples of failure elements, the lower bound of the Ditlevsen bounds provides a good estimation of the system reliability.

$$P_{f_{sys}}^3 = 7.2627 \times 10^{-3} \rightarrow \beta_s^3 = 2.4455$$

Table 42: System reliability of the structure for different levels

Level	Reliability index (RI)	Failure Probability
0	2.3855	0.00853
1	2.3855	0.00853
2	2.3901	0.00842
3	2.4455	0.00723